EMAGIC

Creator SL

MIDI SEQUENCE &
SCORE EDIT / PRINTOUT SYSTEM
SOFT LINK
MULTI APPLICATION MANAGER
FOR ATARI ST SERIES

PART 2

THIS MANUAL INCLUDES ALL CHAPTERS RELATED TO CREATOR SL

EMAGIC

Notator SL/Creator SL

Manual – Volume 2

MIDI Sequence/Score Edit Printout Management System
Soft Link Multi Application Manager

for

Atari ST Serie

Chapter 15

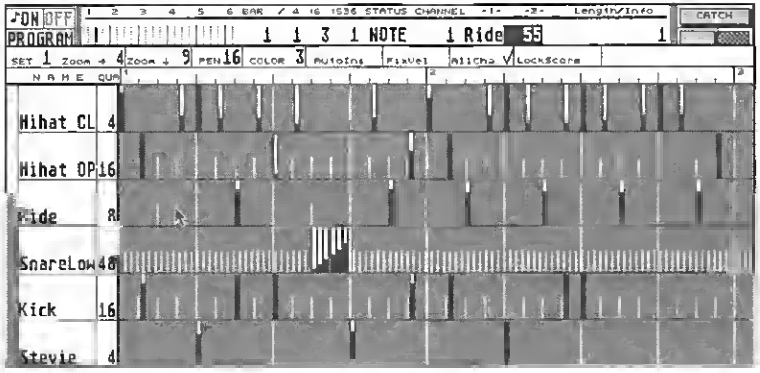
Hyper Edit

15

1. Introduction

Hyper Edit has a huge variety of functions, ranging from the creating and editing of percussion tracks, to the graphical drawing of event values on the screen with the mouse for controlling, say, MIDI Volume. Hyper Edit makes it very easy to edit one particular type of event within a track even if it is mixed in amongst other events.

Whilst the score editor and the matrix editor are really designed to display notes' pitches and positions, Hyper Edit can be configured to single out particular events (such as a particular note or a particular MIDI Controller) and graphically display those events' positions together with their values (such as Velocity or Controller amount).



There are eight »Sets«, each containing up to sixteen »Instruments«.

A Hyper Edit »Instrument« shows the data values of a specific type of event in the form of vertical beams, with time being shown along the horizontal axis.

You can assign any note or any other MIDI event or Pseudo (»P_USER«) event to an »*Instrument*«. As a rule, an »*Instrument*« will be set to single out a specific value within the event's »first data byte« (eg pitch C4). Alternatively, it could display *all* the values of the »first data byte«, such as all the notes on MIDI Channel 5.

If you use all eight Sets, a total of 128 MIDI events (including Pseudo events) can be singled out by the Instruments.

As soon as you have defined a Hyper Instrument, the selected events will be shown on a grid in the form of vertical beams, looking rather like miniature »faders«. The beams show where the events are time-wise, and their vertical heights represent the value, normally of the »second data byte« (eg in the case of notes, their Velocity). By dragging the beam up and down using the mouse, you can easily change the value. The width of the beam is set by »PEN« in the Set parameters.

How much of an instrument you see horizontally and vertically on the screen is determined by the two »Zoom« parameters.

Hyper Edit allows you to add, delete and copy events. It also contains special conversion functions (Convert, Set Conversion, Do Groove, Do Hihat) which make it possible to change the meaning of Sets, Instruments and much more.

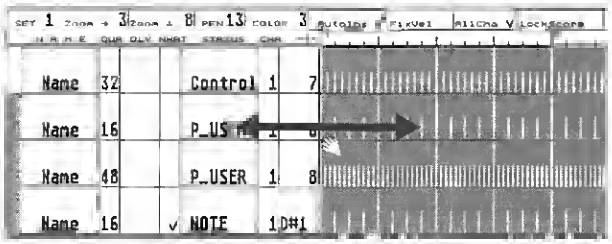
The whole of event editor, minus the matrix editor (event list, score editor and Hyper Edit) is active at the same time. Editing one part of the event editor will directly affect the display of all the other parts.

Points to bear in mind while learning:

- There are two basic ways to use Hyper Edit: as a tool for the creation of music (especially drum tracks, Control events etc); and as a tool for the analyzing of a track's contents.
- You must define an Instrument before it will display the desired type of event. The event will be displayed when its »STATUS« and »first data byte« are matched by the Instrument; a further condition can be that the Instrument must match the event's Channel (if »AllCha« is disabled in the Set parameters).
- Hyper Edit is always showing the contents of one track, not several tracks, but because you are able to view each note pitch etc in a separate Instrument, it is as if you were viewing more than one track.

2. Switching on; choosing amount of display

Press ›J‹ to activate Hyper Edit within the event editor. Alternatively, click »Hyper Display« in the »Edit« menu.



Change the size of the window by dragging the top *horizontal* dividing line (and/or the bottom horizontal line in Notator).

Enter the settings you require in the Instrument parameters on the lefthand side. These parameters can then be covered by the editor itself, with only the Instrument's name remaining visible.

To change the ratio between the parameter area and the Hyper Edit grid, simply use the mouse to drag the *vertical* dividing line left or right. Press ›Alternate-J‹ to toggle between the last ratio you selected and the maximum possible size of the parameter area.

You can also select the vertical window *within* Hyper Edit by dragging the »scroll box« up and down the lefthand edge of the screen within the »scroll bar«. If you click inside this speckled scroll bar, Hyper Edit will jump to the next page.

Tip: »MIDI Out« in the »Edit« menu or pressing ›Shift-O‹ will switch the MIDI output on and off in Hyper Edit as well.

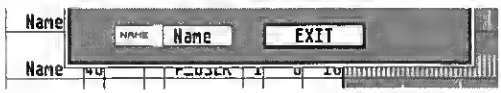
3. Time positioning

All the positioning operations in the event editor also apply to Hyper edit. Moving in time bar-by-bar can be done using the ›(‹ and ›)‹ keys in the calculator keypad.

4. Defining Hyper Instruments

These are the Instrument parameters:

NAME (Instrument name) right-click where the Instrument says »name« to open its Functions window.



You may type in the new name directly (use ›Esc‹ first to clear the naming line). The name is purely for your information and serves no other function. The Functions window provides further copy and conversion functions *(see section 9 »Special Hyper Edit functions«).*

QUA (Quantization) the value corresponds to the MIDI quantization of the track. Here, however, it determines the resolution of the time grid in which new data can be *entered*: the grid visually changes according to the current quantization, making it easier to have an overview of what is going on and to locate specific time positions.

Note: Changing the »QUA« value does not quantize events that are already in the track, only events that you manually enter with the mouse.

DLY (event delay/advance) this parameter, expressed in clock »pulses«, has a similar function to the »DELAY« track parameter on the main page, with this important difference: instead of moving *the whole track* »later« or »earlier«, Hyper delay affects *only that Instrument*. Hyper delay alters an Instrument's event time positions, and the result will be seen in the event list. Any new data you add will already be delayed/advanced by the amount selected.

The »Delay in ms« option also relates to the »DLY« value.

Tip: »DLY« is the only Hyper Edit parameter which will alter existing data if you use it. Because of this, your clicking must be a little slower to allow the program to alter the data: try using »mouse-as-slider« or the ›Control‹ & mouse feature for speed.

»Quick Increment«: if you click in the »DLY« box about a centimetre from the right edge, the values will jump in musical fractions (1/96th, 1/64th etc) for quick changes.

HHAT (»Hihat mode«) for notes that you enter with the mouse into Hyper Edit: if you tick two (or more) neighbouring Hyper Instruments, this creates a »mutual exclusivity«

within this group of Instruments on the same beat. It stops Instruments in a »HHAT« group being active at the same time. For example, if you enter an open hihat at particular time position, any closed hihat which may be there already will be deleted.

Note: »HHAT« groups within a Set can include as many Instruments as you like (max sixteen). More than one »HHAT« group can be defined for each Set: to keep these separate, there needs to be at least one Instrument between two groups without a HHAT tick.

The following three Instrument parameters are the ones that actually do the work of determining what events in the track the Instrument will display. An event has to fulfill all the conditions set in these parameters to be displayed (»CHA« excepted – see below):

STATUS select the status of event that the Instrument is to display. Any MIDI or P_USER status can be selected here. If you select »*undef*«, then it is neutral or »undefined« (see below).

CHA (MIDI Channel) select the MIDI Channel of the events that the Instrument is to display. Any of the sixteen Channels can be selected here. If »AllCha« is tikked in the Set parameters, this Channel value will be ignored and will not be treated as a condition of display.

–1– (»first data byte«) select the »first data byte« of the events that the Instrument is to display. For example, where the status is a note, its pitch will be defined here; for a Control event, the Control number, and so on. If you select »ALL«, the Instrument will display all notes irrespective of their pitch, all Controllers irrespective of their number etc.

With »Control« status, value »1« represents modulation data, value »7« for Volume changes, and so on.

With »normal« Aftertouch (»C-Press«), Program Changes and Pitchbend data, this »–1–« parameter serves no function.

Where the status is a P_USER event, you can select the P_USER number; for example, P_USER 1 allows you to display/create tempo variations.

(For details of what constitutes a »first data byte« etc, see Chapters 8 »Types Of Event« and 9 »Editing in the Event Editor«).

LEN (defined note length) this dictates the length of a note if you enter the note manually with the mouse into the Instrument; it is expressed in clock »pulses«. Set this to a value such as sixteen, but not 0 or 1: this optimizes MIDI timing and avoids having a Note On and Note Off message being sent at almost the same time.

4.1 Automatic definition of Instruments

To define an Instrument automatically:

- Make sure at least one Hyper Instrument is undefined (Status »*undef*«)
- Click the desired event in the event list
- Press ›Shift-J‹.
- Alternatively drag the event from the event list and drop it on top of the undefined Hyper Instrument.

This transfers the parameter values of the current event into the parameter line of the first undefined Instrument. The cursor will then jump to the next event in the list. If there are related events (eg the different notes of a drum machine) one after the other in the event list, you can quickly define these different events as Instruments simply by repeatedly pressing ›Shift-J‹.

For example:

- Put Creator/Notator into record mode and record, say, sixteen percussion sounds.
- Activate an unused Hyper Set (eg Set 8) in the event editor.
- Ensure all the »STATUS« parameters contain the word »*undef*«.
- Jump to the beginning of the event list (press ›Clr Home‹).
- Keep pressing ›Shift-J‹ until you reach the end of the list. The notes in the event editor will now automatically be transferred into the Instrument parameters.
- You should name the different Hyper Instruments straight away: right-click in the »NAME« column and enter a name (eg »Snare«) in the Functions window which appears.

Note: Creator/Notator will make sure that no Instruments are defined more than once. If you do try to define the same event as an Instrument more than once, you will get the message: »This Event is defined as Instrument«. This will also happen if you have reached the end of the event list and defined the last event more than once.

If you press ›Shift-J‹ without having any »undefined« Instruments in the current Set, you will get the message: »*Please mark any instrument as undefined*«.

5. Editing events

You can change the values of existing events in the Hyper Edit grid by dragging the beams vertically with the left mouse button.

If »AutoIns« (Auto Insert) is switched on, events will automatically be entered if you left-click individual empty spaces in the Hyper Edit grid or move the mouse horizontally keeping the left mouse button pressed.

If »AutoIns« is switched off, events will only be entered if you hold the ›Shift‹ key at the same time as clicking.

With »AutoIns« switched off, you can still change the *value* of the events (the height of the beams), but cannot enter new ones.

Be careful where you click the mouse button if you have »AutoIns« switched on.

To delete events, click the beams with the right-hand mouse button.

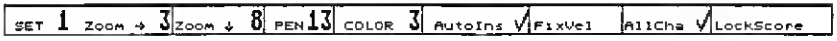
Inserting and deleting can be carried out with a »wiping« motion to the left or right, keeping the mouse button pressed. The distances between the data as it is entered is defined by the Instrument's quantization. You can change the quantization at will within an Instrument to produce different »event densities« for different sections of the music.

6. Global Set parameters

The Set parameters are immediately underneath the dividing line between the event list and Hyper Edit: they contain parameters that apply globally to a whole Set.

These Set parameters are separately programmable for each Set, and can be changed at any time.

The current Set number is specified under »SET«.



6.1 Horizontal and vertical zoom

Immediately next to the Set number, you will see two »Zoom« parameters, one with an arrow pointing to the right, one with an arrow pointing downwards.

The horizontal zoom determines the number of bars visible on the screen at the same time.

The vertical zoom determines the number of Hyper Instruments visible at one time. This obviously also depends on the size of the Hyper Edit window you have selected.

The size of Instruments is automatically selected so that Instruments are always fully visible.

Note: not all sixteen Hyper Instruments can be shown on the screen at the same time. Use the scroll bar on the left of the screen to reveal the areas you want to be visible. If you click in the grey speckled »scroll bar« above or below the white »scroll box«, Hyper Edit will jump in the appropriate direction to the next »page« of Instruments.

6.2 »PEN« and »COLOR«

»PEN« defines the width of the graphical beams used to display the individual events (value: 1 to 16).

The exact time position of an event is represented by the *lefthand* edge of the beam.

The beam widens to the right to a greater or lesser extent, depending on the »PEN« value. A wider beam is easier to manipulate.

The »COLOR« parameter determines the colour and shape of the Hyper Edit beam and the background:

Color 1 white background. Height of black beam depends on second data value.

Color 2 white background. Black beam varies in height, with the remaining area at the top of the beam being grey.

Color 3 grey background. Black beam varies in height, with the remaining area at the top of the beam being white.

Color 4 grey background. Black beam varies in height.

If you use Hyper Edit as a drum editor, »COLOR 3« is recommended to emphasize the individual events, whilst »COLOR 1« helps to enhance continuous Controllers such as Volume.

6.3 »Auto Insert«

When *»AutoIns«* is switched on, a left-click enters data.

If you move the mouse keeping the right button pressed, current data will be deleted irrespective of the mode you are in.

6.4 »Fixed Velocity«

If *»FixVel«* is switched on, you can insert notes with a fixed velocity value. The default velocity is that of the last velocity you edited in the event list or in Hyper Edit.

If »AutoIns« mode is switched off and you move the mouse over note beams keeping the left mouse button pressed, notes in that area will be given the same fixed velocity.

6.5 »All Channels«

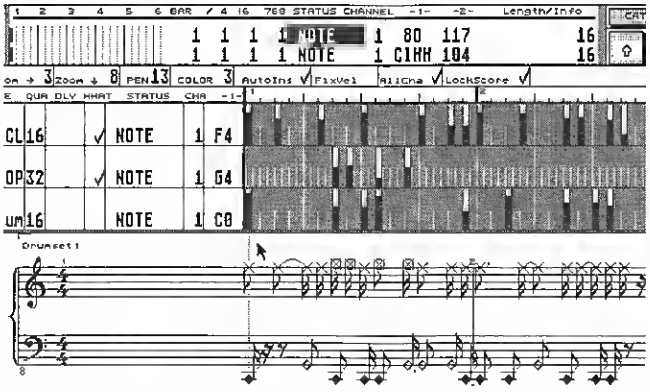
An Instrument only displays the events that match the Instrument's »STATUS« and »–1–« parameters. You can go further than this:

»AllCha« decides whether an Instrument should display all those events regardless of their Channels (»AllCha« on), or only events whose Channel matches the value entered in the »CHA« parameter column (»AllCha« off).

(For those in the know, you could compare this with a sort of »visual« version of the MIDI modes »Omni On« and »Omni Off«.)

6.6 »Lock Score«

Switching on »Lock Score« forces Notator to »synchronize« its score editor visually to Hyper Edit, bar-to-bar.



This makes it easier to edit the dynamics and at the same time see where you are in the music.

Note: »Lock Score« mode overrides Notator's automatic display formatting: the score editor display format will be based on the horizontal zoom. This makes it possible to have »nonsensical« settings of the horizontal zoom such as squeezing twenty sixteenth notes into half a centimetre!

7. Mouse/keyboard combinations

To avoid constantly having to switch »AutoIns« mode on and off and to carry out special functions, there are combinations of the mouse and the computer keyboard.

With »AutoIns« mode switched OFF, the following options are available:

7.1 »Insert«

Left mouse button plus ›Shift‹ key: this is the same as having »AutoIns« mode switched on; events are automatically inserted in empty spaces. Existing events' values can be changed.

7.2 »Additional insert«

Left mouse button plus ›Alternate‹ key: data is inserted in spaces where there are no events. The values of existing events cannot be changed.

Note: after insertion, events become »old events«: you cannot subsequently change their value while ›Alternate‹ is pressed.

7.3 »Delete on quantize positions«

Right mouse button plus ›Alternate‹ key: only those events which are exactly on the Instrument's quantize steps are deleted.

For example: where there are sixteenths in a track, then if you have set 1/8 quantization, only the first of each pair of sixteenths will be deleted, creating 1/16th syncopations.

7.4 Global in/decrease

Mouse plus »Control« key: if you keep the ›Control‹ key pressed, the value of all the selected Instrument's events can be increased or decreased together by moving the mouse forwards or backwards (do not click!); events outside the display area will also be changed.

It is enough just to point inside an Instrument's area and press ›Control‹: no clikking is necessary.

Note: changes are limited to a maximum of 127 and a minimum of 0 or 1. Any changes exceeding these limits are therefore not fully reversible.

8. Examples of Hyper Edit

8.1. Editing a track's velocity

Record a piano track using a velocity-sensitive sound and define an Instrument as follows:

NAME – Velocity
QUA – 16
DLY – 0
HHAT – - - -
STATUS – NOTE
CHA – 1
–1– – ALL
LEN – 16

Make sure »AllCha« is active in the Set parameters and »FixVel« and »AutoIns« are switched off. If you have Notator, switch on »Lock Score«. Try out different »PEN« and »COLOR« settings.

After you have chosen the zoom settings, use the mouse to »paint« the dynamics on the existing events.

8.2 Generating and editing non-note events

The following example uses MIDI Control 7 which controls MIDI Volume.

Play a track with long chords. In Hyper Edit, define an Instrument as follows:

NAME – Volume
QUA – 96
DLY – 0
HHAT – - - -
STATUS – Control
CHA – 1 (or the Channel of the synth's track)
–1– – 7 (Control 7, MIDI-Volume)
LEN – (irrelevant)

Although it is logical to display MIDI notes as individual, separate beams, a continuous curve or »waveform« is more suitable for displaying continuous non-note operations. You should therefore use the following settings in the »Set information bar«:

Horiz. Zoom – 7
Verti. Zoom – 18
Colour – 3
Pen – 16
FixVel – Inactive
AllCha – Inactive
AutoIns – Active
LockScore – Inactive

If you draw a volume curve (keeping the left mouse button pressed), this will appear black and unbroken against the white background.

Tip: in practize, it is rarely necessary to use a »QUA« value of »96« for these non-note events, though the end result looks prettier! It is normally enough to enter Volume etc changes at 1/16ths or 1/32nds intervals. Of course, once the desired volume has been reached, do not continue entering events, since the last event remains in force until a new one is entered (that's the whole principle of MIDI!). Do not be over-generous with non-note events – they use up memory.

8.3 Hyper Edit as a »drum editor«

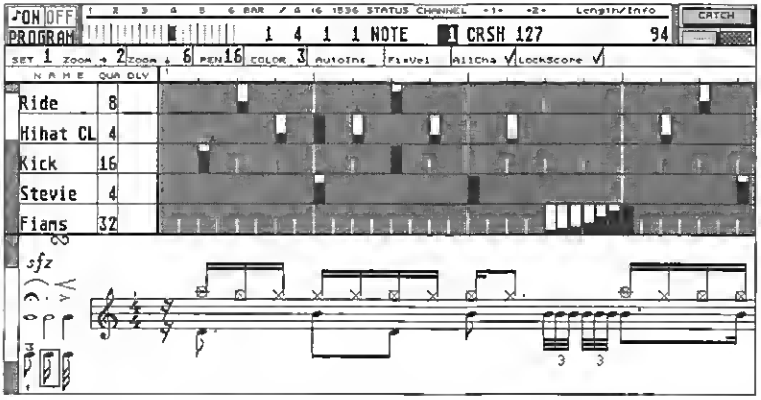
The following example shows the quick way to define a Hyper Set in order to edit an existing drum track.

- To use Hyper Edit efficiently, you need to end up with all the drum notes in one track. If you normally record your drum sounds on different tracks, merge them together to one track. Suppose you have a track containing a kickdrum, snare, closed hihat, open hihat, crash and toms 1, 2, 3.
- Enter this track's event editor and select a free Hyper Set whose Instruments are still »undefined«.
- Find a closed hihat event in the event list. Holding it by the name »NOTE«, drag the event downwards and drop it anywhere onto the first Instrument (alternatively, press ‹Shift-J›). The first Hyper Instrument is now defined and should be named (right-click in the »NAME« column).
- Do the same with the remaining notes and Instruments, using, for example, the following order:

Closed hihat → Instrument line 1
Open hihat → Instrument line 2
Crash → Instrument line 3
Kickdrum → Instrument line 4
Snare → Instrument line 5
Toms 1 to 3 → Instrument lines 6 to 8

- Make a »HHAT« group for the two hihats and crash (tick the »HHAT« column in all three Instruments). This ensures that only one of these Instruments plays at any one time when notes are mouse-entered.
- Make a »HHAT« group for the three toms.
- For each instrument, give the desired grid quantization value in the »QUA« column, in case you wish to add notes with the mouse.

Tip: in Notator, combining a drum-editing Hyper Set with the »Drum Mapping« function for the score makes editing even easier: switch the Hyper Set's »Lock Score« on (see Chapter 11: »Score Display«, section 26 »Percussion notation«).



9. Special Hyper Edit functions

9.1 Copying an Instrument's data

One Instrument's events can be copied to another Instrument, for example to double a snaredrum with a handclap.

To do this, drag the »source Instrument« on top of the »destination Instrument« with the left mouse button, dragging by the name area (similar to copying a track on the main page).

This will copy the events only: their pitch, Channel and even status will be changed to conform to the receiving Instrument's parameters.

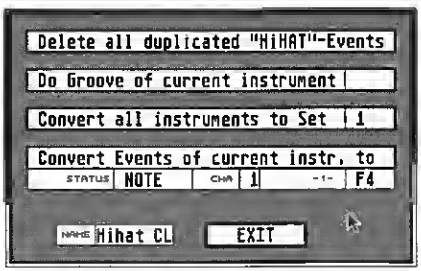
9.2 Swapping Instruments' positions

You can swap the positions of Instruments, to make, for example, a bass drum and open hihat appear directly one above the other.

To do this, drag the »source Instrument« on top of the »destination Instrument« with the left mouse button (drag it by the name area) and press the right mouse button before releasing the left mouse button (a »legato« click, the same way as for moving a track on the main page). This will swap the Instruments' display positions without affecting their data.

9.3 Hyper Edit functions window

If you right-click the Instrument's name area, a window will open containing a number of functions.



Each function is activated by clicking the field bearing its name. »EXIT« will take you out of the window without activating any function.

9.3.1 »Delete all duplicated »HiHat« events«

Where you have Instruments showing events that were recorded in realtime, and you wish to retrospectively delete another Instrument's events that occupy the same time position, ensure the Instruments are neighbours, enable their »HHAT« parameters (ticks in the »HHAT« column) and click the »Deleting all duplicated »HiHat« events« field: priority is given to the Instrument from which this function is carried out.

For example, events in the »open Hihat« Instrument will delete events in the »closed Hihat« Instrument which are located on the same time position in the track.

9.3.2 »Do Groove of current Instrument«

This function allows you to quantize or »groove« a Hyper Instrument separately. Realtime recordings within a track can be processed after recording in many different ways including preset grooves, user grooves and so on. Simply set the desired value and click »Do Groove of current instrument«.

Note that Grooves »16 A« and »8 A« are equivalent to quantize values »16« and »8« (straight 1/16ths and 1/8ths).

This makes it easy to manage percussion recordings *within a track*, by creating separate groove settings and separate delays (DLY) for each Hyper Instrument, as well as creating separate velocity changes using the »Control-Mouse« combination.

9.3.3 »Convert events of current Instrument to ...«

This function allows you to convert all the events of an Instrument into any other events. There are separate settings for Status, MIDI Channel and where necessary the »first data byte«. To do this, open the Instrument's functions window whose events you want to convert; enter the desired parameters (Status/Channel/–1–) in the window and click »Converting events of ...«.

For example, if you want to convert a bass drum into a shaker: right-click the name of the bass drum Instrument and set the shaker note in the window under »–1–«.

Note: do not forget to change the name of the Instrument as well.

The »Transform« function also performs a similar job *(see Chapter 24 »Transform«)*.

9.3.4 »Convert all Instruments to Set ...«

This function allows you to convert the events of all Hyper Instruments in a Set at one pass according to the definitions of another Set.

The most common use of this function is allocating different drum computers, multitimbral modules, samplers etc. according to their note allocations.

The following are important:

- The key allocations of two different MIDI devices must already be defined in two different Sets.
- Both Sets must use the same order when assigning a drum instrument to a Hyper Instrument (eg kick = Instrument 1, snare = Instrument 2, etc).

For example, suppose you want to adapt a drum track originally done for a Roland D-110 to an Emu Proteus module. If the allocation for the D-110 is in Set 1 and that for the Proteus is in Set 8, you need to do the following:

– Activate Hyper Edit for the drum track
– Select Set 1
– Open the Instrument's functions window (right-click on Instrument name)
– Set »Convert all Instruments to Set 8« and click this field.

The drum track will now sound with the Proteus' sounds.

Chapter 16

Matrix Editor

16

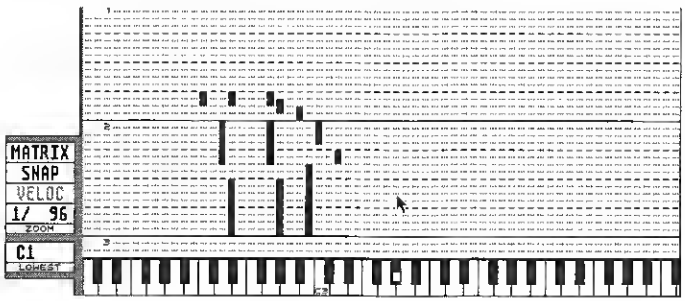
1. The basics

[K] In the event editor, pressing ›K‹ or selecting »MATRIX display« in the »Edit« menu or clicking the »MATRIX« icon on the left of the event editor screen activates the matrix editor, a graphical, interactive display of notes in the form of beams.

It is useful as a general purpose graphical editor, with special emphasis on the editing of note lengths.

Pitch is displayed from left to right, the time position of the note is the top end of the beam, and the length of the note is the length of the beam.

Clicking a beam in the MATRIX editor makes the event list cursor jump to that note event in the list. The matrix editor is simultaneously active with the event list: interacting with a note in one editor will affect the other.



The matrix editor will not run simultaneously with the score editor or Hyper Edit.

2. Dividing the screen

The dividing line between the matrix editor and event list is moveable: click and drag the line up or down.

The event list can be reduced to one line of information: this will always show the »current« event.

The ›(‹ and ›)‹ keys in the calculator keypad scroll the matrix display bar-by-bar; simultaneously pressing ›Shift‹ scrolls the display in smaller steps.

3. Zoom In, Zoom Out

Click the »ZOOM« value to control the resolution of the matrix editor. You may zoom in and out of the display: »1/768« means that one pixel (one small point of the screen display) represents a 1/768 note; in »1/16«, a pixel represents a 1/16 note.

The black, solid lines are bar lines; the grey, dashed lines represent the beats; the dotted, dashed lines represent the divisions of the beat (the »display format« seen in the »information bar« along the top of the screen) divisions, eg 1/16ths.

Keys ›\‹ and ›Shift-\‹ zoom the display from the ST keyboard (USA: ›Z‹ and ›Shift-Z‹).

4. Entering and deleting notes; »SNAP« mode

Left-clicking the matrix plays a note at the pitch of the click. A »dot« will follow the mouse's movements on the screen keyboard along the bottom of the screen. The keyboard is five octaves wide; its base note can be changed by clicking the box marked »LOWEST«.

Right-clicking the matrix when »VELOC« is disabled enters a note at the precise location of the tip of the mouse pointer. The note length, MIDI Channel and velocity all depend on the last note to have been clicked or edited.

When »SNAP« is enabled, beams will snap in to the nearest division, ie they will »auto-quantize«.

Right-clicking a beam will delete it.

5. Transposing and moving notes

Clicking and dragging a beam left or right will transpose a note.

Clicking and dragging a beam up/down will move a note in time.

The first direction you move in dictates whether you are transposing or moving. Adding the right mouse button allows you to move in any direction. You will hear the note move.

Creator automatically prevents you from »nesting« notes, that is, from forcing two notes sharing the same pitch and Channel to overlap.

6. Editing note lengths

Clicking and dragging a beam up and down by its *lower end* allows you to change the note length.

With very short notes, click and hold the mouse button in the area behind the beam, then move the pointer up into the beam – this engages the lower end which can then be moved.

7. Editing velocity

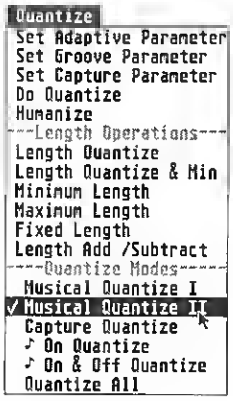
When »VELOC« is enabled (velocity edit), clicking and scrolling the mouse pointer on a beam with the left or right mouse button increases or decreases the note's velocity.

Chapter 17

Quantize, Groove Design, Adaptive Groove

17

Quantize, Groove Design and Adaptive Groove are important functions in Creator and Notator, as they change the time positions of events (notes in particular) according to pre-programmed criteria. They are used especially to automatically correct human timing deficiencies, but they can be used »creatively«, such as using Groove Design to give a »swing« feel to music that was originally played in strict time.



All three error-correcting functions are available as »track parameters« for each track: the QUANTIZE parameter includes the Adaptive Groove values if clicked with the left mouse button.

QUANTIZE is the classic method of auto-correcting timing: it has a range of preset error-correcting templates.

GROOVE DESIGN has two sides to it:

1. It provides further *preset* error-correcting templates based on »swing« feels and other timings.

2. It allows the user to *freely design* his/her own error-correcting templates.

With ADAPTIVE GROOVE, the Program automatically selects its error-correction from a range of pre-defined templates.

GROOVE DESIGN is able to do anything that QUANTIZE can do, and ADAPTIVE GROOVE can do anything that GROOVE DESIGN can do.

For any error-correction to work properly, you must have recorded the music in time with the metronome, and ideally have played as accurately as possible.

Otherwise, the program cannot guess where your beat is, and cannot apply a timing template to the music. The more accurate your original playing, the more you can do with the error-correction.

We recommend you start by familiarizing yourself with QUANTIZE and the preset GROOVES (eg 16 B and 16 C). After that, you are ready to use ADAPTIVE GROOVE.

1. Quantize

1.1 The basics

Quantization is the program's way of adjusting the time position of events to conform (with a range of strictness) to a preset »timing template«: those recorded notes that do not fall exactly on the quantization steps (eg 1/16ths) of the timing template are shifted so that they do.

Quantization, because of the amount of calculation involved, has to *alter* the timing of events. Of all the other track parameters such as »TRANSPOSE«, which are »playback« parameters that are calculated in realtime as the data is played back, QUANTIZE and GROOVE are the exceptions *(see »Appendix«, section 1.1)*. This, however, will not affect you in practice, since:

QUANTIZE and GROOVE are reversible, can be applied before, during or after recording (with results you hear immediately) and generally behave like the other track parameters.

You will notice that, even though QUANTIZE and GROOVE act like the other track parameters, their effect on time positions *can be seen in the event editor*, something not possible with the other track parameters. This is so you can make the correct musical decisions without guesswork, since timing is so important.

F In the main page, pressing ›F‹ (»Fix Quantize«) will fix the time positions of the events in the track at their current (quantized or grooved) positions *(see Chapter 6 »Tracks«, section 4.3 »Fix Quantize/Groove«)*.

1.2 How to ...

A track is quantized simply by scrolling to the desired QUANTIZE value in the track parameter box. Each track can have a different QUANTIZE value.

The values are: 1/4, 1/6, 1/8, 1/12, 1/16, 1/24, 1/32, 1/48, 1/64, 1/96.

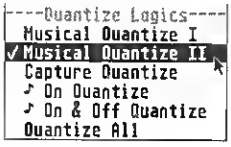
The »1/768« or »1/1536« settings represent the track in its un-quantized state (sometimes called »realtime«).

Clicking the »DO QUANTIZE« icon on the main page or in the »Quantize« menu, or pressing ›Q‹ quantizes the current track with the QUANTIZE or ADAPTIVE GROOVE value last used in any track.

A quantize value given to an empty track will lead Creator/Notator to »automatically quantize« after recording in that track.

This means there is no need to quantize after each recording, as quantizing occurs automatically when ending a recording via STOP or START. Apart from the time-saving aspect, there is no difference between quantizing before or after a recording. If recording in CYCLE mode, quantizing will occur automatically after each round *(see Chapter 7 »Recording«)*.

1.3 Quantize mode



In the »Quantize« menu there is a range of different quantize modes. The program defaults to »Musical Quantize II« mode.

The quantization mode is selected by clicking in the menu; the current mode has a tick by it. Only one mode is available at a time.

Changing mode does not affect already-recorded tracks, only future recordings. However, using »DO QUANTIZE« on an already-recorded track with the new mode in force will re-quantize the track with the new mode.

The following modes are available:

1.3.1 »Quantize all«

The only mode which quantizes every type of event: Notes, Pitch Bend, Controllers etc. Its use is probably limited to the creation of special effects.

1.3.2 »Note On and Off quantize«

Quantizes Note On and Note Off events independently of each other. This can create some drastic note *length* alterations, which can appear as either very long or very short notes (some so short that they will, depending on the sound, sound »swallowed«). This quantization effect can be used to produce »slap-bass« or »picked-guitar« results.

1.3.3 »Note On quantize«

Quantizes Note On events only, does not move Note Offs: will therefore change the note length slightly.

1.3.4 »Musical Quantize I«

Quantizes Note On events, keeping the note length intact.

1.3.5 »Musical Quantize II«

The default quantization for all types of music in all situations, since it alters the »musicality« of the music as little as possible: it quantizes like »Musical Quantize I« but the danger of the mode misinterpretating the timing is almost zero because the musical context is taken into account.

The decision as to which position a note receives is influenced both by the note itself and the surrounding notes. Notator/Creator will recognize systematic timing characteristics, interpret and evaluate them and react accordingly. For instance, where there is any doubt, »Musical Quantize II« will tend to advance the notes of someone who always tends to play behind the beat. Chords are treated as a whole. In practice, »Musical Quantize II« is very inconspicuous and the quantized result will virtually always be as you had expected.

1.3.6 »Capture Quantize«

Allows you to set a »Capture Range« and »Capture Strength« on »Musical Quantize II« mode. In other words, you can quantize a track, but set limits as to what gets corrected, and by how much. Like all quantize modes, Capture Quantize is reversible.

Click on »Set Capture Parameters« in the »Quantize« menu:

»CAPTURE RANGE« gives a range either side of the ideal quantize step, expressed in 1/768th notes. If a note lies inside the range, it will be moved; if outside, it will remain where it is, unquantized.

Note: a Range of »24« means 24 pulses before and 24 pulses after the »ideal« quantization step. »24« is therefore the same as a normal 1/16th quantization since the area between two quantization steps is fully covered by two neighbouring notes' Ranges (1/16th is 48 pulses). A Range of »10« at a quantization of 1/16ths would therefore leave an unquantized area of 28 pulses between each 1/16th, etc.

Example:

Capture Range 96 – standard 1/4 note quantization.
Capture Range 48 – standard 1/8 note quantization.

Capture Range 24 – standard 1/16 note quantization.

»CAPTURE STRENGTH« determines by how much a note will move if it is within the Capture Range, expressed as a percentage.

– A 100% setting will move the note all the way to the nearest quantization step, which is what »Musical Quantize II« does.

– A 50% setting will move the note only half the time-distance toward the quantization step.

– A 1% setting will barely move the note: it will remain virtually unquantized.

So Capture Strength allows the correct »dose« to be given, ranging from »no quantization at all« to »completely quantized«.

Clicking »DO QUANTIZE« in the »Set Capture Parameters« window will immediately carry out the quantization in the current track and change the quantize mode (for future quantization) to »Capture Quantize«. »EXIT« leaves the window without activating »Capture Quantize« mode.

Examples:

Quantize – any value
Capture Range – maximum amount
Capture Strength – 100%

This is equivalent to Musical Quantize II.

Quantize – 1/4
Capture Range – 24
Capture Strength – 100%

Here, any notes that are within 24 pulses of quarter note positions will be moved all the way to them. Other notes will remain unaffected. This can be useful in solos, where you have total timing freedom, except that the beats are tightened up.

Quantize – 1/16
Capture Range – 48 or more
Capture Strength – 75%

Here, all the notes come within the Range, as »48« is greater than the distance between two 1/16th notes. However, notes will only be moved part of the distance to the »ideal« 1/16ths. In a solo, all the notes will be tightened up, but not so much as to be »inhuman«.

2. Groove Design

2.1 The basics

Historically, sequencer quantization has been used as a simple way of correcting the timing errors of recorded notes so as to give the music a »tighter«, more »polished« feel.

Because the timing of computers is so precise, a whole generation of musicians is growing up, whose timing expectations are quite different from the days before these advanced sequencers were around. Modern »high-tech« musicians have come to regard quantized timing as the norm , and human timing as somehow below standard, such that even technically-proficient musicians have felt that they needed to quantize their playing via a computer in order to be able to compete. The down-side of this searching for perfection is the loss of much of the human feel that went into the original recording of the notes.

Groove Design presents two ways of dealing with this problem of »machine versus man«:

Preset grooves firstly, it allows you to call up preset micro-timing templates (»grooves«), such as all kinds of swing-timings, quintuplets, septuplets, combined error-correction to 1/16ths and 1/24ths simultaneously, etc.

User-defined grooves secondly, it allows you to take tracks you have recorded and use them to error-correct the timing of other tracks. The computer's analysis of the timing characteristics of the template track means that all the tracks that are »looking at« the template will play together in perfect time with each other, without the necessity of imposing a »blanket« error-correction.

Grooves can be applied or removed during recording or playback. Switching between grooves or between GROOVE and QUANTIZE will always take the original timings of the notes as a starting point.

The note lengths (gate time) will normally remain unaltered. If the program detects the fact that a Groove parameter has forced two notes of the same pitch and MIDI Channel to overlap, the length of the first note will be correspondingly reduced.

2.2 Preset grooves

Scrolling the »GROOVE« track parameter with the right mouse button allows you to access a range of preset grooves.

You may alter the groove parameter at any time, even during recording and playback, and any changes made will have an immediate effect on the data.

Pressing ›G‹ (»DO GROOVE«) applies the groove value last used in any track to the current track.

A groove value given to an empty track will lead Creator/Notator to »automatically groove« after recording in that track.

The following preset grooves are available:

»16 A« – same as Musical Quantize II, 1/16th note (50%)
»16 B« – Swing 54.1%
»16 C« – 58.3%
»16 D« – 62.5%
»16 E« – 66.6%
»16 F« – 70.8%
»8 A« – same as Musical Quantize II, 1/8th note (50%)
»8 B« – Swing 54.1%
»8 C« – 58.3%
»8 D« – 62.5%
»8 E« – 66.6%
»8 F« – 70.8%
»8 to 12« – combined error-correction of 1/8 and 1/8 triplets
»1216« – combined error-correction of 1/16 and 1/8 triplets
»five« – Quintuplet error-correction
»sept« – Septuplet error-correction
»nine« – Nine-tuplet error-correction
»Free« – This template is a special case. It provides a »floating« swing template so that you are not stuck with using just one of the preset templates. It stops the offbeats of a swung pair of notes from being error-corrected if they fall in the area, for instance, between the 1/16th offbeat and the 3rd triplet note (ie between »1 1 2 1« and »1 1 2 17«: the »Free« zone). This means that you can be fairly free with the degree of swing you play: your »swing factor« will not be lost. Because some of your notes are not being corrected (in the interests of »feel«), it is best used for solo runs etc, rather than for multi-track laying.

The two combined error-correction grooves are less flexible than their Adaptive Groove *(see below)* equivalents, because they do not take the musical context into account.

Tip 1: use »Fix« where you are swinging a track that was played in strict time (see Chapter 6 »Tracks«, section 4.3).

Tip 2: to groove all the tracks of a pattern, press ›Shift‹ while changing the »GROOVE« track parameter *(see Chapter 6, section 2).*

2.3 User-defined grooves

Scrolling the »GROOVE« track parameter with the left mouse button allows you to access a range of user-defined grooves.

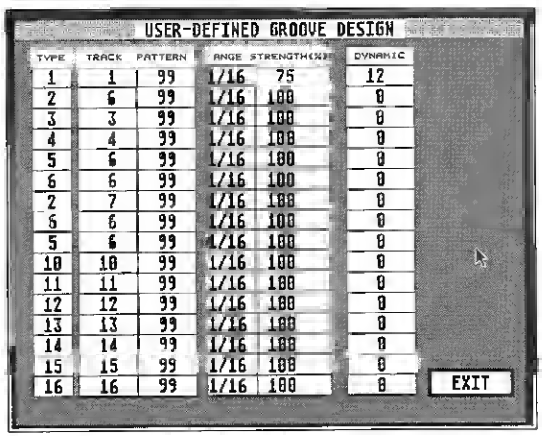
Groove Design allows you to design your own timing-correction templates. The templates are based on the time positions of notes in a track, so where there is a note, there is a potential timing-correction step. A template track is created either by defining a track recorded in realtime as a template, or by manually entering notes into a track.

»Set Groove Parameters« in the »Quantize« menu is where you can determine which track is to be used as timing-correcting template for the other tracks: in one of the sixteen »Groove Type« lines, enter the pattern and track of the template and any Capture Range and Strength.

When a track has been defined as an timing-correcting template, any track that is instructed to »look at« this track (via its »Groove Type«) will have its notes moved to time positions that as far as possible coincide with those of the template.

The sixteen user-defined templates are activated by scrolling to the »Us 1« to »Us16« values in the »GROOVE« track parameter: »Us 1« is »TYPE 1« in the »Groove Design« window, etc.

The template track can be as long as you wish. It can even be »looped«, which means that the »template« part will consist of the selected number of beats which will loop from the top of the track to the selected loop-point.



The »Groove Design« window has sixteen lines, so allowing up to sixteen user-defined »Groove« templates to be active at the same time, ie rhythmical/dynamic templates whose source can be any track.

The first column »TYPE« shows the Groove number (US 1 to US 16). The second and third columns (»TRACK« and »PATTERN«) determine the source of the Groove. The fourth and fifth columns determine, as in »Capture Quantize«, the »Range« and »Strength« of the template.

»RANGE« gives a range either side of each template note, expressed in 1/768th notes. If a »slave« note lies inside the range, it will be moved; if outside, it will remain where it is, unquantized.

»STRENGTH« determines by how much a »slave« note will move if it is within the Capture Range, expressed as a percentage. A 100% setting will move the note all the way to the nearest template note. A 50% setting will move the note only half the time-distance toward the template note. A 1% setting will barely move the note: it will remain virtually unquantized.

So »Strength« allows the correct »dose« to be given, ranging from »no timing-correction at all« to »completely corrected«.

In the above example, pattern 99 is shown as containing sixteen tracks, each of which is acting as a user-defined template. This is one way of organizing a collection of template tracks; the advantage here is that they can be collectively saved to/loaded from disk via »Save/Load Pattern«. You may also like to save such a pattern of favourite templates in your »AUTOLOAD.SON« *(see Chapter 28 »Data Management«, section 2.3 »Autoload Song«)*.

It can make sense to sort the templates in order of increasing effect, so that you can compare the, say, »weakest« timing corrections starting at Us 1 against the »strongest« in Us16.

Tip: use the »Control«&mouse feature to save time when scrolling from, say, »Us 1« to »Us16«, as you can switch over in one leap (see Chapter 3 »Concepts«, section 1.2).

2.3.1 Examples of user-defined grooves

Example 1: User-defined swings (eighths): you yourself may program swing times into a user-defined template track; you are not restricted to using the preset swings of A to F. First, you must create the template track

– enter the event editor of track 1, pattern 99 via »EDIT«.

– enter two notes (pitch irrelevant, since all we are interested in is time positions). The display »Format« is assumed to be 1/16.

1 1 1 1 any pitch
1 1 3 x any pitch

Where »x« is you could enter any value you please : a »1« would mean no swing at all (50%). Try entering a »9«:

– LOOP the track after one beat: »1«.

This way, the template loop is repeated every beat which means that it can time-correct a track of any length.

– In the »Set Groove Parameters« window, enter:

Type 1 = pattern 99, track 1, Range 48, Strength 100%.

You have defined »GROOVE Us 1«. Now whenever you give a track a GROOVE value »Us 1«, the track will be error-corrected according to the template track. In this particular case, the swing corresponds to preset Groove »8 C«.

Variation: for a 1/16 swing feel, you would need to program four notes into the first beat of the template track:

1 1 1 1 any pitch
1 1 2 x any pitch
1 1 3 1 any pitch
1 1 4 x any pitch

For a normal type of swing feel the 1/768 represented by »x« would be of the same value, eg »5«. But you can go much further in your experimentation.

Conversion Table: 1/768ths – Swing Percentages

	50%	54.1%	58.3%	62.5%	66.6%	70.8%	75%
1/8 swing:	1	9	17	25	33	41	49
1/16 swing:	1	5	9	13	17	21	25

Tip 1: an 1/8 swing template track can be changed to 1/16 with the use of »Double Speed«, and vice versa with the use of »Half Speed«.

Tip 2: the percent (and per thousandth) values may also be read directly in the event list if »Position in ms« in the »Flags« menu is enabled at a tempo reading of 60 for 1/8th notes, or tempo 30 for 1/16th notes. At these tempi, an event's time position can be exactly translated into swing percentages, eg event position 00.546 ms is exactly

54.6%. Once you have entered the values, you can of course change the tempo to whatever you like.

Example 2: suppose you want an eight-bar verse to have a slight swing:

1 1 1 **1**, **1** 1 2 5, 1 1 3 1, 1 1 4 5 etc,

but the chorus needs something stronger:

1 1 1 1, 1 1 2 9, **1** 1 3 1, 1 1 4 9 etc,

the passage between the two can be made to flow if in the last (eighth) bar of the verse, the template track is given:

8 **1** 1 **1**, 8 1 2 **5**, 8 1 3 **1**, 8 1 4 **5**,
8 2 1 **1**, 8 2 2 **6**, 8 2 3 **1**, 8 2 4 **6**,
8 3 1 **1**, 8 3 2 **7**, 8 3 3 1, 8 3 4 **7**,
8 4 1 **1**, 8 4 2 **8**, 8 4 3 1, 8 4 4 **8**

in other words, an increasing amount of swing with each beat.

In this example, two or four events in the template track with their LOOP values will not work. You would need a template track as long as the song itself. Using »Segment Copy« makes this easier to achieve: the four notes whose positions are 1 1 1 1, 1 1 2 5, 1 1 3 1, 1 1 4 5 make up a quarter note, so in »Segment Copy« *(see Chapter 18)* enter:

»Source:**1** 1 1 1 to 1 2 1 **1**«,
»To Position:1 2 1 **1**«,
»Number of Copies:31«.

This gives you the first verse, including that eighth bar which you now must edit as above.

Tip 3: until now, we have been delaying the off-beats. By delaying or pushing all the beats over long sections, you can achieve a pushing or dragging type of feel in the music.

Tip 4: the character of a »groove« can be reinforced further if there is an element of Velocity dynamic in the track. One tends to heavily accent the beat; with percussion instruments like the hihat, clave, shaker, cowbell etc it makes more sense to stress the weaker beats (see Dynamic Groove, below).

Example 3: since any track can be used as a template track, even those tracks which have already been modified by GROOVE in some way can be used as template tracks.

This principle makes sense where, for instance, you wish to use the realtime playing of a real (!) drummer on his MIDI pads as your basis for timing-correcting the remaining tracks in the song: however, where this drummer has included pauses and

drum rolls in his playing, this might confuse the program in its calculations. This is how you might reach a satisfactory result:

- Make up a long track consisting of quantized 1/16th notes using SEGMENT COPY (the pitch is irrelevant).
- Make a copy of the drum track (or maybe just the best bit), erase the rolls and make this the template track via the »Set Groove Parameters« window (eg Type 1/Us 1). Where necessary use the range and strength parameters to focus in on the desired beats.
- set GROOVE »Us 1« in the 1/16ths track.
- now use that (ex-)1/16th note track as the definitive template for the other tracks in the song by making it Type 2 in the »Set Groove Parameters« window« and setting »Us 2« in the GROOVE parameters of the other tracks.

That way, any pauses in the original drum track will not interrupt the playing since there is now a note on every sixteenth to keep the flow.

2.4 Dynamic Groove

The note velocities of a template track can be forced onto other »slave« tracks. Insert the Type number of the template track (Us 1 to 16) into the »slave« track's GROOVE parameter and press the ›V‹ key to »DO DYNAMIC GROOVE« (simply scrolling to the desired template value will not alter the velocity, just the timing).

How much the template track will affect the slave tracks' velocities when ›V‹ is pressed is determined by the setting in the Groove window's »DYNAMIC« column. ›V‹ has no effect on the time position of notes.

The window's DYNAMIC values range from 0 (no effect) to 15 (maximum effect). Dynamic Groove is event-altering and so can be reversed only by UNDO (until the next event-altering function is carried out). Since Dynamic Groove takes the latest version as its starting point, pressing ›V‹ progressively increases the effect, so try using lower values in the window and pressing ›V‹ more than once till satisfied.

Since the act of scrolling to a desired template activates the time-position-altering side of Groove Design, if a pure Dynamic Groove is desired without affecting the timing, set the track's STRENGTH value to 0% in the »Set Groove Parameters« window.

3. Adaptive Groove

3.1 The basics

»Adaptive Groove« is a correction strategy for note positions, which brings together all the QUANTIZE and GROOVE templates in one function.

Whereas »Musical Quantize II« and »Groove Design« settings apply to the whole track, »Adaptive Groove« will independently recognize and analyze the structure of the recorded music, and will then quantize or groove segments of the track separately, depending on the requirements of the music.

Any combination of the quantize templates (»4« to »96«), preset grooves (»Free« to »16A«) and user-grooves (»Us 1« to »Us16«) is possible, up to eight templates at a time in what is called an »Adaptive Groove Set«.

Eight of these »Adaptive Groove Sets« are available per song.

3.2 Applying an Adaptive Groove Set to a track

The eight Adaptive Groove Sets can be called up, one at a time, by scrolling the «QUANTIZE« track parameter with the left mouse button, starting from the »no quantization« setting «768«/«1536«.

Each Set is user-nameable in the »Adaptive Groove Settings« window *(see section 3.3.3 below)*.

Notator and Creator come with seven pre-programmed Adaptive Groove Sets:

1624 – 1/16ths and 1/16th triplets are recognized.

16C+ – 1/16ths and 1/16th triplets are recognized, syncopated 1/16ths will be »swung« according to preset groove »16C«.

1632 – 1/16th and 1/32nds are recognized.

All – 1/8ths triplets, 1/16ths, 1/16ths triplets and 1/32nds are recognized.

Joba – 1/8ths triplets, 1/16ths, 1/16ths triplets and 1/32nds are recognized. Syncopated 1/16ths will be left uncorrected, according to preset groove »Free«.

1216 – 1/8ths triplets and 1/16ths are recognized.

1224 – 1/8ths triplets and 1/16ths triplets are recognized.

As with user-defined Groove Design, any other settings are possible: *see section 2 »Groove Design« above.*

The last QUANTIZE template or Adaptive Groove Set to be chosen can be applied to other tracks by left-clicking the »DO QUANTIZE« icon or pressing ›Q‹.

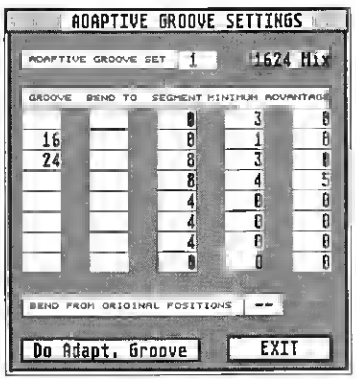
A right-click on »DO QUANTIZE« returns the template in the QUANTIZE track parameter (whether it is a quantize template or Adaptive Groove Set) back to »768« or »1536« (no time correction). The same happens if you click the QUANTIZE parameter about one centimetre from the right edge of the track parameter box; use either mouse button, depending on whether the current template is quantize or Adaptive Groove.

Tip: because Adaptive Groove is a wide-ranging correction strategy, initial calculation times can take longer than the simpler quantization. So, to select the desired Set in the »QUANTIZE« track parameter, use the ›Control‹&mouse feature to jump straight to it (see Chapter 3 »Concepts«, section 1.2).

3.3 Defining the Adaptive Groove Sets

3.3.1. The Adaptive Groove Settings window

To define the Adaptive Groove Sets, you use the Adaptive Groove Settings window: click »Set Adaptive Parameters« in the »Quantize« menu, or press ›Alternate-A‹.

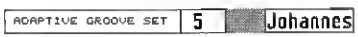


3.3.2 Set number

Click in the box at the end of the »ADAPTIVE GROOVE SET« line to select one of the eight Sets' numbers. Each complete window is a Set.

3.3.3 Set name

Each Set can be given a name of up to eight characters; the first four characters are



visible in the »QUANTIZE« track parameter on the main page (in other words, you select the Adaptive Groove Set by the name you give it rather than by its number).

3.3.4 »GROOVE«

In the »GROOVE« column, you decide which quantizing and groove templates should prevail during the automatic analysis of segments of the recorded music.

Up to eight quantize and groove templates are available simultaneously in the column.

Any combination of the quantize templates (»4« to »96«), preset grooves (»Free« to »16A«) and user-grooves (»Us 1« to »Us16«) is possible. Clicking the left half of a »GROOVE« box with the left or the right mouse button deletes a template.

Tip: when experimenting for the first time, start by using just two templates to avoid getting confused.

Example 1 – GROOVE

Line 1 – 16
Line 2 – 24

Result: combined correction of 1/16ths and 1/16th triplets.

This corresponds to the pre-programmed Set »1624«.

Example 2 – GROOVE

Line 1 – 16
Line 2 – 32

Result: Combined correction of 1/16ths and 1/32nds.

This corresponds to the pre-programmed Set »1632«.

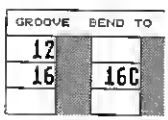
Example 3 – GROOVE

Line 1 – 16 C
Line 2 – 24

Result: combined correction of »swung« 1/16ths and metrically-exact 1/16th triplets.

3.3.5 »BEND TO ...«

In the »BEND TO ...« column you can decide whether there should be a second correction template, especially important when the first template is a groove.



When the »GROOVE« column templates have done their correction, there can be a final correction according to what is programmed in the »BEND TO ...« column.

The same templates are available in the »BEND TO ...« column as in the »GROOVE« one.

Example:

GROOVE	BEND TO
16	16 C
24	(blank) empty box

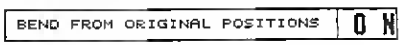
This corresponds to the pre-programmed Set »16C+24«.

Any segment which was analyzed as being a 1/16th by the »GROOVE« template will be »bent to« or transformed into »swung« 1/16ths.

Segments which are given the 1/24th »GROOVE« will not be affected by »BEND TO ...«.

3.3.5.1 »BEND FROM ORIGINAL POSITIONS ON/OFF«

In the »BEND FROM ORIGINAL POSITIONS ON/OFF« line, you can decide whether, before there is any processing by the »BEND TO ...« column, the data should first be corrected according to the »GROOVE« column.



»BEND FROM ORIGINAL POSITIONS: ON«

This corresponds to the normal timing-correction logic for a track, with calculations based on the notes' original time positions.

»BEND FROM ORIGINAL POSITIONS: (OFF)«

Here, further correction calculations are based, not on the notes' original positions, but on their new positions acquired as a result of event-altering (like »Fix Quantize« *(Chapter 6 »Tracks«, section 4.3)* or »Process Data« *(see Chapter 6, section 4.2)*. Here, however, the notes' original positions are remembered.

Tip: for realtime-recorded music which contains different amounts of »swing« combined with full triplets, you may find better results when »BEND FROM ORIGINAL POSITIONS« is switched off.

3.3.6 Regional template structure

3.3.6.1 »SEGMENT«

In the »SEGMENT« column, each template can be given a segment length ranging from a quarter note to a sixteenth. This determines over what period a template is active, within which a particular correction takes place.

Example:

GROOVE	BEND TO	SEGMENT
16	--	8
24	--	8

If, having analyzed the structure of the music, »Adaptive Groove« decides to use a 1/16th template, then in the above example, the corresponding segments within the region of an 1/8th note will be corrected (eg the first and fourth eighths of a bar).

If, having analyzed the structure of the music, »Adaptive Groove« decides to use a 1/24th template, then in the above example, the corresponding segments within the region of an 1/8th note will be corrected (eg the third and fifth eighths of a bar).

Important! The program splits the music up into the segments which are defined in the »SEGMENT« column of the Adaptive Groove Set. Segments not included here remain unavailable.

If segments of differing lengths are selected, then there should be one or more »alternatives« to the smallest value.

Example:

GROOVE	BEND TO	SEGMENT
16	—	8
16	—	16
32	—	16

If »Adaptive Groove« decides, for example, to correct the first 1/16th segment of a bar with a 1/32nd template (Line 3), the additional line 2 (1/16th segment with a 1/16th template) acts as an alternative for the second 1/16th segment of the bar. Without this alternative, it might also be corrected with a 1/32nd template.

In general: several small segments automatically make up an alternative if they fit into a larger segment.

Tip: the »SEGMENT« column can allow nonsensical settings: if the »GROOVE« value is 1/12th, and the »SEGMENT« value 1/16th, the program cannot process this. To recognize a 1/12th template, one or more 1/8th triplets must be set in the »SEGMENT« column.

3.3.6.2 »MINIMUM«

Within an »Adaptive Groove« segment, there must be a minimum number of notes available before a decision in favour of the template can take place: the »MINIMUM« column defines this minimum number of notes within the segment.

»MINIMUM« works closely together with »SEGMENT« and is an important factor in the choice of the template. If »MINIMUM« is set to »1«, it has no effect.

Example:

GROOVE	BEND TO	SEGMENT	MINIMUM
16	—	8	1
24	—	8	3

These two lines carry out a combined correction of 1/16ths and 1/16th triplets. However, the 1/16th triplets will only be accepted for correction if the complete set of three notes is in the segment.

If the second line in the above example had a »2« in the »MINIMUM« column, »incomplete« triplets would be accepted as well (though there must be at least two per 1/8th segment).

Tip: you can also have settings that do not make sense in the »MINIMUM« column. If the minimum number of notes will not fit in the segment, the template cannot be chosen (for example: »GROOVE« 1/24, »SEGMENT« 1/8, »MINIMUM« 5).

Where there is any doubt, the program will choose any one of the available »GROOVE« templates.

3.3.7 »ADVANTAGE«

The value (0 to 10) in the »ADVANTAGE« column gives the amount of advantage a »GROOVE« template has over the other templates. »GROOVES« with larger »ADVANTAGE« values are more likely to influence the correction than others.

As a rule, there are practical advantages in giving smaller »GROOVE« values fewer chances, so that any timing problems in the music can be caught and dealt with.

Example:

Groove	Bend To	Segment	Minimum	Advantage
16	—	8	1	10
24	—	8	2	7
32	—	8	4	6

In this example, the 1/16th correction has the greatest advantage. Although incomplete 1/16th triplets will be accepted, they must have been very accurately played. Only four 1/32nd notes in a row will be recognized, and only if they were played accurately.

Tip: all »ADVANTAGE« settings are relative to each other: if »10« is set in each line, it is the same as having »0« in each line.

3.4 »Do Adaptive Groove«

The current Adaptive Groove Set can be applied directly to the current track by clicking the »Do Adapt. Groove« icon. Clicking »EXIT« or pressing ›Return‹ quits the window without actioning the function.

Alternatively, select one of the Adaptive Groove Sets in the »QUANTIZE« track parameter.

The last QUANTIZE template or Adaptive Groove Set to be chosen can be applied to other tracks by left-clicking the »DO QUANTIZE« icon or pressing ›Q‹.

3.5 Managing the Adaptive Groove Sets

Although Adaptive Groove Sets can be »tailor-made« for pieces of music which you have recorded in realtime, it makes sense to have »tried-and-tested« Sets available at any time: save these in your »AUTOLOAD.SON« *(see Chapter 28 »Data Management«, section 2.3)*.

The eight Sets can also be loaded into existing songs via »Load System« in the »File« menu *(Chapter 28, section 3)*.

4. Note Length Quantize

The lengths of notes can be quantized without this affecting their positions *(see Chapter 19 »Event-Altering Functions«)*.

Chapter 18

Copy, Merge, Insert, Cut, Delete, Move, Swap

18

1. Mouse operations within a pattern

1.1 Track copy

Within a pattern, to duplicate a track, drag it (by its name) to a new *empty* destination track and release the mouse button.

1.2 Track merge

Within a pattern, to merge two tracks, drag one (by its name) on top of a new *recorded* destination track and release the mouse button. You will be asked to confirm or cancel the operation.

Where any of the two tracks' parameters are the same (eg both are set to Channel 5), those track parameters will remain as they were; where any of the tracks' parameters are different, the events of both tracks will be altered instead according to what each track's parameter was *(see Chapter 6 »Tracks«, section 4.4)*.

1.3 Track exchange

Within a pattern, to swap the position of two tracks, drag one (by its name) on top of a new *recorded* destination track and release the mouse button. You will be asked to confirm or cancel the operation.

1.4 Track move

Within a pattern, to move a track to a new position, drag it (by its name) to a new *empty* destination track; *without releasing the left mouse button*, depress the right mouse button as well, then keeping the right mouse button depressed, release the left mouse button.

1.5 Track replace

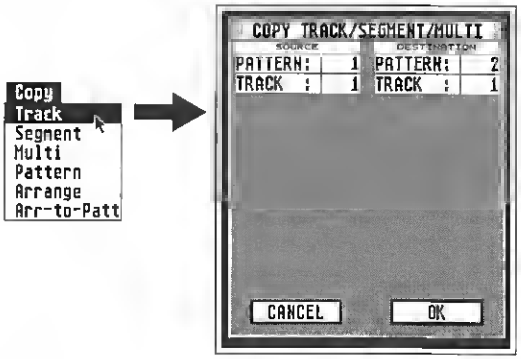
Within a pattern, to replace a track with another track, drag it (by its name) on top of the track to be replaced; *without releasing the left mouse button*, depress the right mouse button as well, then keeping the right mouse button depressed, release the left mouse button. You will be asked to confirm or cancel the operation.

1.6 Track delete

Dragging a recorded track to the left hand side of the screen deletes it (or press »Backspace«). However, you can get it back by clicking the »UNDO« icon (until the next delete or other event-altering process).

2. Copying a track

2.1 »Copy« menu: track



You may copy the events of a complete track including its track parameters (the »source« track) to an empty track in a different pattern (the »destination« track) with »Track« in the »Copy« menu.

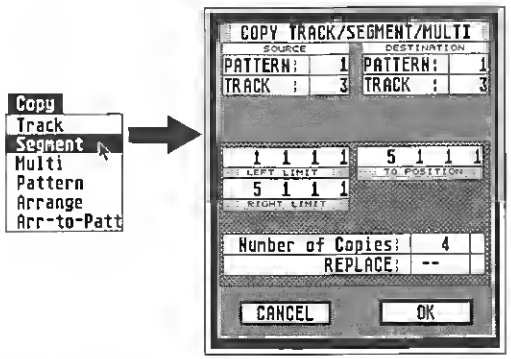
If the destination track already contains data, you will be asked if you wish to MERGE the two tracks into the destination track or cancel.

2.2 Copying a track via keystroke

Click the track you wish to copy then press ›Shift-C‹. Now select the pattern, click the desired track position and press ›Return‹. This will copy the track across.

This is a flexible alternative to the »Copy« menu, because you can select the track you want to copy first before looking for its destination position.

3. »Copy« menu: segment



You may copy the events of part (»segment«) of a »source« track to a »destination« track with »Segment« in the »Copy« menu.

For the source track, select the segment's left (inclusive) and right (exclusive) limits, and select the destination track and the left limit or start position of the segment's destination.

If »REPLACE« is switched off (»--«), and the destination track already contains data, you will be MERGING the segment into the destination track.

If »REPLACE« is switched »ON«, you will be erasing the destination track's data at that point and replacing it with the segment.

The source segment may be copied many times end-to-end by selecting a »Number of Copies« value, so »filling« a track or part of track with the chosen segment.

The source track can also be the destination track.

If you are in the event editor, you can define the left limit by clicking that event and pressing the ›F1‹ key, and define the right limit by clicking the last note of the segment and pressing ›F2‹.

When you go to »segment copy« (eg by pressing ›Alternate-C‹), the limits will be ready. *Note:* because the right limit is always exclusive, it will actually show a position one pulse more than the last event, so that the last event is included in the segment.

Tip: before using Segment Copy with the »fill« feature, check whether the »LOOP« track parameter wouldn't be better: it uses no extra memory and is quicker to use, but you can only loop from 1 1 1 1 (see Chapter 6 »Tracks«, section 3.6).

3.1 Examples of segment copy

Example 1: bars 21, 22, 23 and 24 of a solo are to be copied to bars 1 to 4 of the same track, and what was already there is to be simultaneously erased:

Source				
Left Limit	21	1	1	1
Right Limit	25	1	1	1
To Position	1	1	1	1
Number of Copies		1		
Replace		ON		

Example 2: when bars 1 and 2 of a drum track are copied as follows the result is one long track of sixteen bars (seven new repetitions and the original):

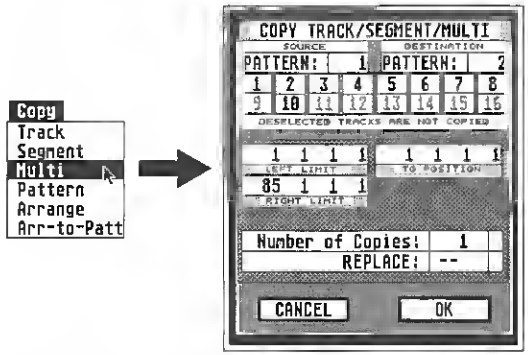
Source				
Left limit	1	1	1	1
Right Limit	3	1	1	1
To Position	3	1	1	1
Number of Copies		7		

Example 3: a typical single 1/16th note repeated »sequencer« line is required over two bars:

- ensure the display format is set to »1/16« (because we are dealing with the third column of the time position – *see Chapter 5 »Positioning«),*
- select an empty track and enter its event editor (press ›E‹)
- enter one note at 1 1 1 1, then do as follows in »segment copy«:

Source				
Left Limit	1	1	1	1
Right Limit	1	1	2	1
To Position	1	1	2	1
Number of Copies		31		

4. »Copy« menu: multicopy

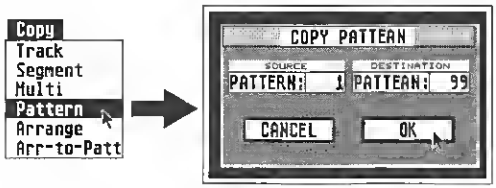


»Multicopy« in the »Copy« menu allows the copying of a segment of up to all sixteen tracks of a pattern simultaneously.

This function uses the same parameters as »segment copy« with the difference that you can switch tracks on and off to determine which segments will be copied simultaneously, which is the same as using »segment copy« several times.

The source and destination segments must have the same track numbers, though these will be in different patterns.

5. »Copy« menu: pattern



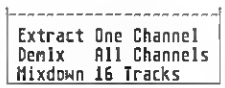
This function allows the copying of a whole pattern with its sixteen tracks to another pattern number.

6. »Mixdown 16 tracks«

»Mixdown 16 tracks« in the »Functions« menu (or press ›Alternate-M‹) merges all the tracks of the current pattern onto the top track.

The MIDI Channels of the individual tracks are forced onto their events, and the »host« track is switched to »original« Channel *(see Chapter 6 »Tracks«, sections 3.1.2 »Channel number »Original« and 4.4 »Merging tracks with differing track parameter values«).*

7. Separating by Channel



Creator and Notator allow the events of up to sixteen Channels to be in one track; this happens, for example, when you have »imported« a complete song from a different sequencer: the whole song will be recorded by Creator/Notator into one track. The following functions allow you to extract events according to their Channels and place them in different tracks.

7.1 »Extract One Channel«

»Extract One Channel« in the »Functions« menu allows you to remove the events of one MIDI Channel in a track and deposit them in any empty track of any pattern.

- Click »Extract One Channel«.
- Under »SOURCE«, select the pattern and track from which you want to extract.
- Under »DESTINATION«, select the pattern and track to which the extracted events are to go.
- Under »MIDI CHANNEL«, select the Channel number of the events to be extracted, then click »OK«/press ›Return‹.

7.2 »Demix All Channels«

»Demix All Channels« in the »Functions« menu (or press ›Alternate-,‹) allows you to separate the events in a track according to their MIDI Channels and deposit them in the empty tracks of the current pattern.

Channel 1 will go to track 1, Channel 2 to track 2 etc.

This function is particularly useful when you want to work on a song that was imported from another sequencer: within a couple of seconds, the song is spread over the current pattern's tracks. Ensure the other tracks are empty before you »Demix«.

8. Separating by »chord Voice«

The following functions (in the »Functions« menu – with the mouse pointer, touch the »... tions« syllable of »Functions«) allow you to split up a track's chords into the chords' con-

stituent notes or »Voices«. For the functions to work properly, the notes in the chords must be on the same pulse. To ensure this is the case, quantize the track first, carry out the function, then de-quantize if needs be.

Note: the functions do not split a track's notes according to their pitch, but according to their relative positions within chords.

8.1 »Extract One Line«

»Extract One Line« extracts a single Voice from chords in the track.

In its dialog box, you can decide whether to extract the highest or the lowest Voice of each chord: the extracted notes will be placed in the next empty track in the pattern.

Repeated use of this function shaves off each line one-by-one to a separate track.

8.2 »Extract All Lines«

»Extract All Lines« splits up all the chords into their constituent Voices and deposits the notes as single Voices in the patterns's remaining empty tracks.

This is especially useful in Notator where you have written, say, a piece for a wind ensemble on your MIDI keyboard, and you then wish to have each musical part on a separate track/stave.

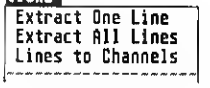
8.3 »Lines To Channels«

»Lines To Channels« splits up all the chords into their constituent Voices by assigning the Voices to different MIDI Channels. No extraction to other tracks takes place.

All the events in the track must be set to Channel 1 before actioning this function.

»Lines To Channels« could be used, say, with Notator's »Polyphonic« display function, which uses the MIDI Channels of events in a track to distinguish between its Voices.

9. Track-based cutting/reversing



9.1 »Cut Inside Locators«

»Cut Inside Locators« (»Functions« menu) deletes all notes and events *in a track* which lie between the left and right locator positions.

9.2 »Cut Outside Locators«

»Cut Outside Locators« (»Functions« menu) deletes all notes and events *in a track* which lie beyond the left and right locator positions.

9.3 »Reverse Inside Locators«

»Reverse Inside Locators« (»Functions« menu) reverses the order of all the notes and other events in a track between the left and right locator positions.

10. Pattern-based cutting/inserting

10.1 »Cut and Move Pattern«

»Cut and Move Pattern« (»Functions« menu) erases all the notes and events between the left and right locator points in all sixteen tracks in the current pattern and simultaneously relocates the data lying beyond the right locator to the left locator position, making a »seamless join«.

Note: UNDO is not available with this feature since more than one track is involved.

10.2 »Insert and Move Pattern«

»Insert and Move Pattern« (»Functions« menu) introduces a blank space between the left and right locator points in all sixteen tracks in the current pattern by pushing back the data lying at the left locator point to the right locator position.

Note: UNDO is not available with this feature since more than one track is involved.

11. Deleting

11.1 »New Pattern«

»New Pattern« in the »File« menu asks whether you want to delete the data in the current pattern, and does so if you click »OK«: this deletes the data in the pattern's sixteen tracks.

11.2 »New Song«

»New Song« in the »File« menu asks you whether you want to »initialize« Creator or Notator, ie wipe all existing music data and start again with a virgin program: make sure your data is saved to disk first. The action of loading in your »AUTOLOAD.SON« has the same effect of giving you a fresh program with the advantage of loading your

favourite default settings. (Loading *any* »*.SON« file erases the existing data anyway, so always check you have saved the existing data before you load.) *(See also Chapter 28 »Data Management«, sections 2 »Load/save song« and 2.3 »Autoload Song«.)*

11.3 »Quit«

Clicking »Quit« at the bottom of the »File« menu asks you whether you want to exit from Creator or Notator and revert to the »GEM desktop«. This destroys all your music data, so make sure you have saved it to disk first. If you are in the Soft Link environment, the only way to quit the program is to switch the computer off; programs in the other partitions, however, are quittable as usual.

11.4 Summary of delete and keep functions

Delete Track (see this Chapter, section 1.6).

»New Arrange« deletes the current arrange list (see Chapter 20 »Arrange Mode«, section 11).

»Delete/Keep Events« in the »Functions« menu (see Chapter 24 »Transform«).

Realtime »Delete/Keep Events« (see Chapter 24 »Transform«).

»Fast Delete/Keep« in the event editor (see Chapter 9 »Editing in the Event Editor«, section 19).

Delete arrange list entry (see Chapter 20, section 4).

»Delete Short Notes« in the »Functions« menu (see Chapter 19 »Event-Altering Functions«, section 11).

Deleting individual events (see Chapter 9 »Editing in the Event Editor«, section 6, Chapter 10 »Score Editor«, section 6.4, Chapter 15 »Hyper Edit«, section 5 and Chapter 16 »Matrix Editor«, section 4).

»Delete File« in the »File« menu (see Chapter 28 »Data Management«, section 7).

There are many other functions that have a deleting side to them (»Cut Inside Locators«, »Cut and Move Pattern« etc). These are dealt with throughout the manual.

Chapter 19

Event-Altering Functions 19

1. What means »event-altering«?

There are, fundamentally, two ways of affecting the data that flows from the MIDI output:

1. By using »playback« parameters that work on the data in realtime as it is being passed through the MIDI output. These include the »track parameters« and arrange mode »pattern parameters«. »Playback« parameters do not alter the original data of the events, but only how they behave outside the computer.
2. By using »event-altering« functions that alter the original data of the events themselves while they are in the computer's RAM.

Also included in the »event-altering« camp are all the changes you make yourself with the mouse, Hyper Edit etc in the event editor.

The effect of using event-altering functions as opposed to »playback« parameters is that you can see the alterations to the events in the event editor.

Event-altering functions include »Transform« *(Chapter 24)*, »Process Data«, »Normalize«, »Fix Quantize« *(all Chapter 6 »Tracks«)* and many others included in this Chapter.

The event-alterations are reversible by using »UNDO« because the original version is kept in a »buffer«, but using »UNDO« to reverse a specific operation is only possible until you carry out the next operation that involves making alterations to events. You cannot »UNDO« operations that involve more than one track.

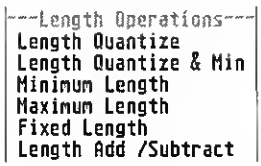
When using any of the following event-altering functions in sections 3 to 13, the name of that function will *replace* the existing track name if the track name begins with a star (ie you have not yet inserted your own name in place of Creator/Notator's default names eg »**OK**« or »* New *«). If the existing track name is one of yours, these function names will not erase it.

2. Forcing track parameter value onto events

The »track parameters« *(Chapter 6 »Tracks«, section 1)* on the main page are normally »playback« parameters: they affect data as it flows through the MIDI Output. However,

their values can be forced onto the events in their track *(see Chapter 6, section 4 »When track parameters can be »event-altering«)*.

3. »Quantize« menu: note lengths



3.1 »Length Quantize«

This quantizes the lengths of notes according to the selected template in the »Length Quantize« window, expressed as bar/beat/sixteenth/pulse; the lengths become complete multiples of this value.

Where an existing note is less than half the length of the »Length Quantize« template, its length is reduced to zero.

This can produce interesting results with certain sounds, such as »picked basses« or »clavinets«.

3.2 »Length Quantize and Mininimum Length«

This has the same effect as »Length Quantize« (see above), but avoids note lengths of zero: the minimum note length is the length of the template.

This can be useful for ensuring chords are of the same length etc.

4. »Quantize« menu: »Minimum Length«

Any note length less than this template will be made longer.

5. »Quantize« menu: »Maximum Length«

Any note length greater than this template will be shortened.

6. »Quantize« menu: »Fixed Length«

With this function you can achieve note lengths which are all exactly the same; avoid too great a length, or notes of the same pitch may overlap.

The »Fast Transform« function in the event list has the same effect: set the desired length in one note and press ›Shift-T‹, which copies that length to all the following notes.

7. »Quantize« menu: »Length Add«

This adds the selected value to the length of notes.

8. »Quantize« menu: »Length Subtract«

This subtracts the selected value from the length of notes.

9. »Functions« menu: »Force Legato«

This alters a track's note lengths by lengthening/shortening notes so that they exactly abut one another, thereby creating a perfect legato effect (depending, of course, on the sound). Simultaneous notes in a chord are given the same length, but only if the notes share the same time position (if necessary, temporarily quantize the track, click »Force Legato«, then un-quantize).

10. »Functions« menu: »Overlap Correction«

This corrects a track if it finds that notes played in a »lazy« legato fashion overlap each other. If such is the case, the previous note's length will be shortened so that its Note Off appears just before the following Note On.

The prevention of overlapping notes is very useful when using an older monophonic synthesizer (fitted with a MIDI interface, of course), since some of these synthesizers need a fresh envelope trigger for each note. Another use is with multi-timbral synthesizers, where »Overlap Correction« allows efficient use of all the synth's voices by stopping »wasteful« overlapping.

Exact chords – several notes on the same time pulse – are recognized, and an automatically-appearing dialog box asks you whether you wish to keep the chords, or delete all the constituent notes except one, depending on whether you want a strict monophony. To make chord notes of an unquantized track simultaneous, it is a good idea to quantize this track temporarily.

11. »Functions« menu: »Delete Short Notes«

This erases all notes whose lengths fall below the template you select here.

Wrong notes (often the result of inaccurate playing) tend to be notes of low velocity and short duration, and these can be easily spotted and eliminated by the program. Another use of this function would be to deal with the relatively common problem generated by guitar synthesisers, where spurious »sympathetic« notes of very short duration are triggered.

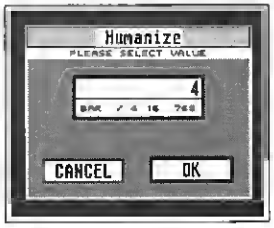
12. »Functions« menu: »Check Duplicated ...«

This erases duplicated (ie identical position, Channel and pitch) notes and events, and simultaneously inserts any missing Note Offs.

»Check ...« is the only way to insert missing Note Offs.

Duplicated notes can occur when, for example, you record while cycling, where the same note can be played more than once as the segment cycles. You can normally hear these duplications as a »phasing« sound, and looking in the event list will confirm this.

13. »Quantize« menu: »Humanize«



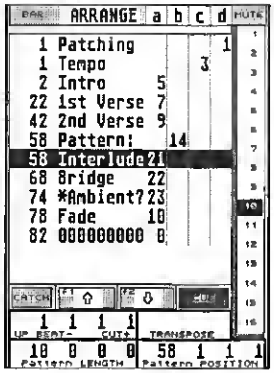
This function adds and subtracts a random amount of time to and from notes' time positions. Notes can thereby receive a certain amount of »quasi-human imprecision«, with the advantage that limits can be set on the randomness.

For example, »8/768ths« would mean that all the events would be randomized up to 4/768ths forward and 4/768ths backward.

Chapter 20

Arrange Mode

20



1. The basics

The arrange mode is a function which allows you to form an arrangement (song, film cue etc) by arranging entries, which represent patterns, in a certain order; it then plays back the entries in the order you have determined. Up to four patterns may play at the same time.

The arrange mode has two display modes, with easy switching between the two:

1. The arrange list, in which the arrange entries take the form of a sequential list of alphanumerical lines *(until section 16, this Chapter describes the workings of the arrange list).*

2. The Graphic Arrange Mode (GAM), which is a graphical representation of the patterns in the form of vertical beams *(see section 16 below).*

For instance, the verses, choruses, bridges, intro and outro of a piece can be recorded independently of each other in different patterns, each with up to 32 tracks, and assembled and played in the desired order in the arrange mode.

Each line of the arrange list or beam of the GAM is called an »entry«: it contains the instruction to the program to play a certain pattern at a certain time, and displays the pattern's name, number and the chain it is playing in.

In this Chapter, we will refer to an »entry« when we are refering to the pattern as part of the structure of the arrange mode, and to a »pattern« at other times.

Tip: think of the arrange mode as a »sequencer for patterns«. The entries in the arrange mode are not recorded material, but simply instructions that tell patterns when they should play.

There are four »chains« (a to d) in which any succession of patterns can be started. In the normal 16-track-per-pattern mode, all four chains can play patterns simultaneously.

The patterns in the chains are independent of each other, and are free to overlap. When you are working with 32-track patterns, chains a and c only are at your disposal.

Tip: there is no connection whatsoever between the chain letters »a« to »d« and the MIDI Output ports »A« to »F« *(see Chapter 6 »Tracks«, section 3.1.3 »MIDI Out ports ...«).*

The »arrange cursor« moves down the list when the sequencer is running, showing which entry is the current one.

Whenever the arrange mode is switched on, get into the habit of selecting patterns by clicking their corresponding entry in the arrange list, rather than manually selecting a pattern number in the pattern window.

An entry in the arrange list consists of a »start-bar« *(see section 3)*, a pattern name *(see section 2)* and a pattern number *(see section 3)*. This pattern number is always in one of the four chains.

An entry in the arrange list only ever contains one pattern at a time, never more, and each entry occupies a separate line. This is so each entry in the list can be given a name. If you have two, three or four patterns starting at the same time, they will occupy two, three or four successive lines: it is each entry's start-bar that tells you exactly when it is starting, not necessarily where it is placed within the list.

With the arrange mode switched on (»ARRANGE« »ON« in the »information bar«), which patterns are played when is determined by their entries' start-bars in the arrange list and corresponding pattern number.

Every single entry in the arrange list can be given a set of »pattern parameters«. Even if *the same* pattern appears more than once in the arrange list, it can be given a different set of pattern parameters each time:

– »UPBEAT/CUT«: you can define a pattern's upbeat or, with »Cut«, define where you are going to start within the pattern.

– »TRANSPOSE«: the pattern as a whole can be transposed.

– »LENGTH«: the length of the entry can be altered.

– »POSITION«: shows the entry's start bar, and allows this to be altered.

– »MUTE«: each pattern can have a different combination of muted tracks.

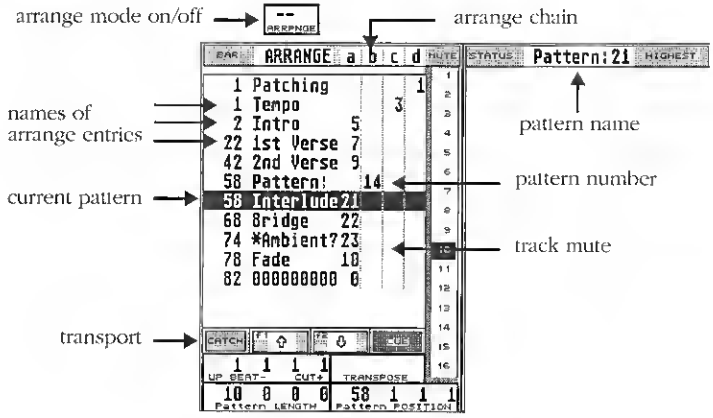
These pattern parameters are, like track parameters, »playback« ones, meaning that they affect data only as it is played through the MIDI outputs: they do not alter data in the computer's RAM. This is why the same pattern can be used more than once in the arrange list and yet still sound quite different each time.

The length of a pattern is determined by the start-bar of the entry that follows it in the same chain. It will stop when the next entry starts in the same chain.

This means that pattern lengths are decided by the succession of entries in a chain.

An entry that has no entry following it in the same chain is of »unlimited« length (»Pattern LENGTH«: »no limit« – *see section 6.4).*

Patterns can be started in each of the arrange chains a, b, c and d, which means that up to four patterns, each with sixteen tracks, can play at the same time.



A song, for example, might be structured as follows:

- Chain »a« has a succession of patterns (intro, verse, chorus, etc).
- Chain »b« can contain a parallel solo in one »long« pattern, running alongside some of the entries in chain »a«.
- Chain »c« can contain the drum tracks, recorded live, in one long pattern, running alongside chains »a« and »b«.
- Chain »d« with its sixteen tracks is still free, and could be used to record RMG data, or anything you like.

The arrange list, with its names and visual arrangement, is self-documenting. The arrange list alters nothing within the patterns' data since all the pattern parameters are »playback« parameters, which means you can change your mind at any stage during composing and arranging.

2. Naming entries in the arrange list

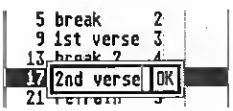
There are two ways of naming entries in the arrange list:

1. An entry in the arrange list can be automatically force-named by naming the pattern window itself: press ‹Shift-N› or double-click where it says »Pattern:XX« above the pattern window.

This is the most flexible way of naming an entry because it is the entry's pattern number that determines the name: the name goes where the pattern number goes. You can override this automatic name by using the following method, but once an entry has been named by the following method, you cannot revert to this automatic one.

So, name your patterns as soon as possible with general names such as »Count«, »Intro«, »Verse«, etc; this will help to show you what is where as your song builds up.

2. An entry in the arrange list can be named by pressing ‹Shift-Esc› with the arrange cursor on it, or double-clicking the entry's name.



This is not the same as the first method. Here, the *entry as a position in the list* is named, irrespective of the pattern number: changing the pattern number has no effect on the name. It is best only used to override the automatic naming of the first method. For example, where the same pattern appears a second time in the list, you could call it »Verse 2«, overriding its original name of »Verse 1«.

Use ›Esc‹, ›Backspace‹, ›Delete‹, left and right cursor arrows as usual when typing in the name.

An unnamed entry in the arrange list defaults to the name »Pattern«.

3. Start-bar and pattern number

The »start-bar« of an entry in the arrange list is to the left of the name. This marks where the downbeat (the »1 1 1 1«) of the pattern is, in the song: in other words, it marks where the entry starts in the arrange list.

The rest of the entry's time position is not shown in the entry line. So, if the start-bar says »8«, it means »start this pattern when the Main Bar Counter says 8 1 1 1«. If the start position of the entry is a beat etc later, the rest of the time position line can be read (and altered) in the »Pattern POSITION« box *(see section 6.1)* below the arrange list.

The start-bar number depends on the selected time signature (eg 4/4). If the signature is altered after an arrange list is installed, the start-bars will change, and so too will the entries› lengths; however what you hear remains the same. You will need to restore the lengths to their original values *(see Chapter 5 »Positioning«, section 1.3 »Time signature«)*. There are about 1350 bars available in 4/4 time.

Start-bars earlier than 1 are possible (minimum: –9), but, in general, you are advised not to start your music before bar 1: in »MIDI Sync« mode, MIDI protocol dictates that an external MIDI »Start« command must come at the beginning of bar 1, so you would lose whatever lay before it. Even the count-in must start at bar 1, which means you music starting at bar 3 or 5, depending on the length of the count-in.

On the other hand, Unitor in »SMPTE Sync« mode is able to handle negative start-bars, so long as the »Sync-Reference« »SONG START« position is made early by the same amount *(see Chapter 27 »Hardware Peripherals: ...«)*.

If you want the music or count-in to start on the 1, but the list is starting earlier, simply move the whole list later by the necessary bars *(see section 6.4 »Pattern LENGTH« below)*.

The pattern number is always to be found in one of the chains a, b, c or d. This determines the chain in which the pattern starts. You can move a pattern across to a different chain by clicking in one of the other chains in the same line.

4. Inserting and deleting entries

4.1 Inserting entries within one chain: single pattern play

To insert entries into the list, you drag them out from each other, like »track copy«: drag an entry downwards in the arrange list to copy it (holding it by its name) *(see also section 4.4 »Song Stop«)*.

Once copied, start-bar, name and pattern number can be changed at any time, so it is irrelevant which entry is used as a »source«, unless you wish to copy a particular pattern for repetition etc: the entries are merely instructions for the »arrange list sequencer« to change patterns by remote-control.

If you drop the entry on top of an existing one, it will be inserted at that position, and the existing one and all the ones after it in the same chain will be moved later by an amount equivalent to the new pattern's length, to make space for it.

Once the new entry is in place, you can change its arrange list length by using its »Pattern LENGTH« box below the list.

If its length is correct, but you want it to start earlier or later, use the previous entry's »Pattern LENGTH« box to shorten or lengthen the previous pattern, for that is what you are saying. Do not touch the new entry's start-bar: you will be able to make it start earlier or later, but then its length will be wrong.

If you drop the entry beyond the last entry in the chain, it will be positioned four bars later than the last entry.

4.2 Inserting entries in more than one chain: multi pattern play

If you need a pattern to run at the same time as other patterns, you can insert a new entry in another chain (say, »b«). The way to do this is by using the ›Insert‹ key.

– Place the arrange cursor on an existing entry whose start-bar is nearest to where you want a new entry to start in the new chain.

Which entry you choose to cursor is not critical but it makes sense to choose one nearest to where you want the new entry to start.

– Press the ›Insert‹ key: this makes a copy of the cursored entry and inserts the new copy above it in the list at the same start-bar, but in the next chain along: the arrange cursor simultaneously moves to the new copy.

It is worth reading that again and doing a few arrange list experiments to ensure you understand that concept! The new entry is inserted so quickly, it is easy to miss what happens.

At this stage, you will have the same entry starting at the same start-bar, but in two chains: a nonsensical situation, since you do not want simultaneous duplicates of your notes! So the next operation after »Insert« is to change the pattern number of the new entry to an empty one, ready to take a recording, and if necessary, change its start-bar to the desired position.

Once it is in a chain, you can move an entry across to a different chain by clicking in one of the other chains in the same cursor line.

Problem: suppose you want a string line to play continuously over the changeover between the verse and bridge patterns in chain »a«. The program does not allow a recording to start one pattern and continue in the next without interruption.

Solution: place the string pattern in the next chain. So, place the cursor on the verse pattern in chain »a«, press ›Insert‹ which duplicates the verse in chain »b« as well. Change the duplicate's pattern number to an empty pattern. Done!

4.3 Deleting entries from the arrange list

To delete an entry from the arrange list, drag it into the right half of the screen and release the mouse button; or, press ›Delete‹ with the arrange cursor on the entry to be deleted.

Deleting an entry has the effect of »telescoping« or shortening the chain that contained the entry by a length equivalent to that of the deleted entry.

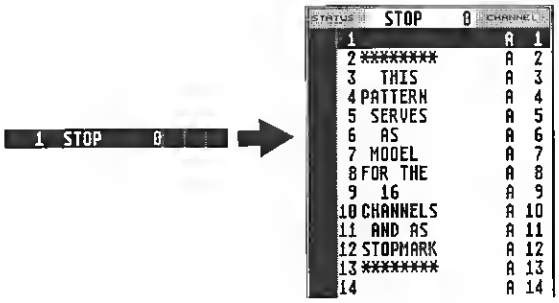
So if you delete an entry that was four bars long in chain »a«, all the following entries will move earlier by four bars in chain »a« to fill the space.

The other chains are not affected by the deletion: so if the other chains are integral to the song, you will have to think about shortening them all by four bars at that point to keep their relationships correct.

To delete an entry without moving the entries after it in the same chain, »click« the entry to be deleted across to an empty chain (one with no entries in it), then delete it.

This preserves the chain which contained the entry exactly as it was, except that the entry that was before the deleted one is now longer by a length equivalent to the deleted one.

4.4 Song Stop



Pattern »0« (zero) acts as a Stop command to the sequencer.

Pattern zero's start-bar marks the point at which the sequencer will stop, so insert an entry containing pattern zero at the end of your arrange list. You will notice that pattern zero is already called »STOP«, so its name in the arrange list will also be »STOP«.

Tip: you may, when building up your arrange list, like to begin by inserting an entry with pattern zero after the initial »default« entry that is always in the list: then drop the other entries (see section 4.1 above) on top of the pattern zero entry. This has the advantage of keeping the copied entries' original lengths.

So, to have a series of 8-bar patterns in a chain (prior to giving them new numbers etc):

- drag out an entry from the top entry and drop it underneath; the top entry's length defaults to »4 0 0 0«, the new one says »No Limit«.
- change the new entry's number to zero;
- place the cursor back on the first entry and change its length to »8 0 0 0« in the »Pattern LENGTH« box below;
- drag out more entries from the top one, dropping them on top of the next entry down. This keeps all the lengths at eight bars, and keeps pattern zero at the end of the list.

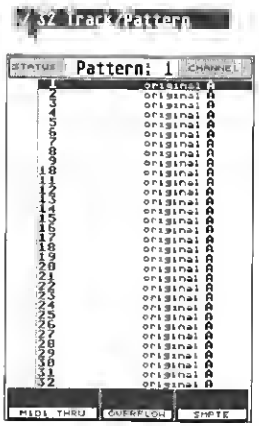
4.5 »32 tracks-per-pattern« mode

With »32 Track/Pattern« ticked in the »Flags« menu, two patterns each containing sixteen tracks become tied together as a »double-pattern«.

The usual sixteen tracks per pattern are often not enough, and so the pattern window can display all 32 tracks of this »double-pattern«.

The 32 Tracks/Pattern mode applies to the whole Song. Since the maximum number of simultaneous tracks possible is 64, the arrange mode chains a and c only should be used.

Tip: make your mind up, before starting a song, whether you wish to work in the 4 chains × 16 tracks or 2 chains × 32 tracks mode.



From now on, we will refer to »patterns« and »double-patterns« to distinguish between 16 and 32-track patterns.

In 32 Tracks/Pattern mode, the double-pattern consists of two neighbouring pattern numbers.

Example: patterns 1 and 2, 11 and 12 etc.

The title line of the double-pattern will display the number of the first pattern only.

Important advice: to avoid confusion, please give double-patterns odd (1, 3, 5 etc), not even, numbers.

Example: if double-pattern 1 is inserted in arrange chain a and double-pattern 2 is inserted in chain c, then tracks 17–32 of double-pattern 1 and tracks 1–16 of double-pattern 2 will play simultaneously: they are identical tracks being played twice over MIDI.

You should also be careful when using double-patterns in arrangements that already contain entries in chains b and d. Try to avoid using 32 Tracks/Pattern mode in your »old« songs: it is bound to lead to trouble.

Despite this, if you wish to reorganize such a song, the best method is to name your patterns, save them with »Save PATTERN« and reload them in their new positions into odd-numbered patterns (having deleted the patterns first, of course, using »New Pattern« in the »File« menu, or the delete function in the »Pattern Overview« function (»Options« menu – *see section 15.1*).

NB, to recap: in the arrange list, for every »master« entry you insert, a second »slave« one is inserted at the same start-bar, in the next chain, and whose pattern number is always the master's +1. *This slave entry is not shown in the arrange list*, however the Graphic Arrange Mode (GAM) will display this by widening the beam.

Important: a few pattern functions work on all 32 tracks of a double-pattern, but others affect the half of the pattern in which the track cursor is situated. The latter situation occurs in functions that depend on the MIDI Channel as a distinguishing criterion, eg:

– Demix all tracks
– Mixdown all tracks
– »Screen recording« of realtime mutes.

In addition:

– New Pattern
– Multicopy
– Display parameters (Notator)

– Changing track parameters using the ›Shift‹ or ›Alternate‹ keys.

The following functions operate on all 32 tracks:

– Cut & move pattern
– Insert & move pattern
– Copy pattern.

Tip for Notator users: use 32 Tracks/Pattern mode to achieve more staves than are offered by a normal pattern. Switching on Full Score mode allows you access to all the double-pattern's staves, for editing, »Page Previewing« and printing etc.

5. Positioning the arrange list

Every entry can be clicked with the mouse.

F1 F2 Function keys ›F1‹ and ›F2‹ move the arrange cursor up and down the list without affecting the Main Bar Counter.

Their icons are also on the screen below the arrange list and can be clicked with the mouse.

With the arrange mode on, clicking »START« starts the sequencer from the beginning of the cursored entry. Clicking »STOP« twice resets the Main Bar Counter and sends the arrange cursor back to the first entry.

(The ›(‹ and ›)‹ keys in the calculator keypad move the arrange cursor up and down the) list, simultaneously loading the left and right locators with the start and end bars of each entry.

These keys allow you to make the sequencer instantly jump from entry to entry while it is playing, and the Main Bar Counter jumps to the start of the selected entry, whereas using the ›F1‹/›F2‹ keys only changes the pattern window display, not what you hear.

5.1 »CATCH« mode

L Clicking the »CATCH« icon below the arrange list or pressing ›L‹ ensures that the arrange cursor scrolls according what the music is doing, ie according to the Main Bar Counter position.

Whenever you start, Catch mode is always automatically enabled.

Clicking anywhere in the arrange list during playback disables Catch mode so that the cursor remains on the current entry even if the music continues.

This allows you to edit etc while the music is playing.

Clicking »CATCH« again makes the cursor catch up with the Main Bar Counter.

»Freeze arrange cursor«: pressing ›Shift-L‹ at any time disables Catch mode. It will remain disabled until you press ›L‹.

In some situations, for example where you are working with long patterns in the chains, it is useful to keep the cursor on the entry you are working on, something not normally possible since Catch mode is usually re-enabled with every Start command.

Compare »Freeze arrange cursor« with »Arrange/Pattern Couple« *(section 13 below)* which allows the arrange cursor to keep moving.

5.2 »CUE«

By keeping the left mouse button depressed on the CUE icon, you can move the arrange cursor up and down the list by sliding the mouse backwards and forwards.

This is a »power« version of the ›F1‹/›F2‹ keys *(see above)*, and is used for moving quickly around the display. While the mouse button remains pressed and the mouse pointer looks like a cross, the mouse pointer can stray from the CUE icon.

By keeping the right mouse button depressed on the CUE icon, you start the sequencer from the beginning of the cursored entry; moving the mouse back and forth changes the tempo, which then returns to where it was when you release the mouse button.

This is a very convenient way of starting and stopping Creator and Notator, and works in an identical way to the CUE function in the event editor.

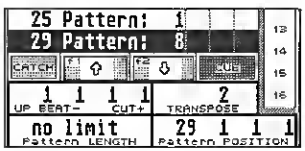
5.3 Arrange position markers

The number keys of the computer's typewriter keypad can be used to store and recall up to ten different arrange cursor positions.

Storing – select the entry and press ›Shift-Alternate-0 ... 9‹
Recalling – ›Alternate-0 ... 9‹

During playback, the sequencer will immediately jump to the recalled entry.

6. Pattern parameters



Each separate entry in the arrange list has a range of parameters, which below (Pattern POSITION, UPBEAT/CUT, TRANSPOSE and LENGTH) and to the right of (MUTE) the arrange list, show the cursored entry's values. They are »playback« parameters that are reversible and non-destructive.

6.1 »Pattern POSITION«

The exact starting position of an entry is displayed in the pattern POSITION box; it can be edited here as well to allow starting times that are not exactly on the bar line.

In practice, the entry's starting time is almost always on the bar, and is determined while you enter it in the arrange list or Graphic Arrange Mode: no need to use the

Pattern Position box. However, in the rare case where an entry starts later than the bar, use the Pattern Position value to move the pattern.

Examples:

1. A chorus should start two beats later than the »17 1 1 1« it defaults to: enter »17 3 1 1« in the Pattern Position box.

2. In VITC Sync, a pattern should start exactly on a cue: activate »Position in Frames« *(see section 6.1.2 below)* and click in the Pattern Position box until the cue's SMPTE time is shown.

3. Introducing time signature changes into a song whose arrange list is already established will result in Pattern Position values that are offset from the bar lines: simply go back through the arrange list, re-establishing the correct lengths till the offset values have gone. Do not touch the Pattern Position values themselves *(see also Chapter 5 »Positioning«, section 1.3 »Time signatures«).*

6.1.1 »Position in ms«

»Position in ms« in the »Flags« menu shows any »Pattern POSITION« in terms of hours, minutes, seconds and milliseconds. The current tempo dictates the actual values.

Arrange list entry start times are given in minutes and seconds, but the complete time position to the nearest millisecond must be read under »Pattern POSITION«.

Bear in mind that tempo changes in the song are not taken into account.

6.1.2 »Position in frames«

»Position in frames« (»Flags« menu) shows any »Pattern POSITION« in terms of SMPTE hours, minutes, seconds, frames and bits. The current »Sync Reference« including all its tempo changes dictates the actual values *(see also Chapter 27 »Hardware Peripherals:...«).*

6.2 »UPBEAT«/»CUT«

»UPBEAT«

Notator/Creator will record every note or other event played into it, even those which are played during a count-in (ie before »1 1 1 1«). These notes played before the official first beat of the song are what is known as an »upbeat«. If an entry in the arrange list makes no allowance for an upbeat, these »count-in notes« will not be played. This is the usual situation and is roughly equivalent to what most sequencers do.

The »UPBEAT« parameter decides the exact changeover point between an entry and the one before it.

Moving this changeover point does *not* alter the position in time of the pattern.

The default »UPBEAT« value »1 1 1 1« switches between the previous pattern and the current one exactly on the »1 1 1 1« of the current pattern. This represents a normal changeover.

By *reducing* the upbeat value of an entry (eg to »0 3 1 1«), the pattern is given more time before its first beat: the previous entry will stop earlier (in this example, by two beats). The current pattern is therefore able to start with its upbeat.

So upbeats are not handled as separate patterns, with all the additional work that entails, but belong within the main pattern, to be used or not as you choose.

Whenever an upbeat for an entry has been allowed for, the »-« sign will appear to the right of its start-bar.

Problem 1: supposing you have had to give a strings track a »DELAY« amount of »–10« to make them sound in time, it means you are making the track play ten pulses earlier than the rest of the pattern: if the first note of the track was on the pattern's downbeat, ie »1 1 1 1«, it is now on »_ 4 4 39«.

You will notice that this first note is not playing: this is because the first note is playing before the »1 1 1 1« but no »upbeat« value has been programmed for it.

Solution: give that entry an »upbeat« value equivalent to the position of that early note (»_ 4 4 39«), and it will be heard again.

Problem 2: suppose you have played an upbeat snare introduction to a drum pattern by starting to play during the last two beats of the count-in. This is a relatively large upbeat and means the track is starting on »_ 3 1 1«.

Without an »upbeat« value for that entry, that snare intro would not be heard. But if you put one in, you will be stopping the previous pattern early by two whole beats, which means probably cutting off some of its music: not a good idea.

Solution: give that entry its »upbeat« value »_ 3 1 1« and then »click« the entry across to the next free chain: this allows the previous pattern to continue playing to its end, but also allows the current pattern to start playing two beats early without stopping any previous pattern. For those two beats, the two patterns are playing concurrently.

Note: in this situation, the previous pattern in the original chain now has a greater »Pattern LENGTH« value because the current pattern is out of the way in another chain. If the previous pattern has any »LOOPED« tracks, they will continue playing throughout the new length, which is not desirable as it would spoil the music of the

current pattern. So remove any loops and use »Segment Copy« to produce tracks of the exact length required.

Tip: even when an upbeat is not musically necessary, it can make sense to have a slight upbeat value of a few pulses: Creator/Notator will sense notes which are still playing at the pattern-changeover point and switch them off with Note Off commands. If the changeover happens, as usual, on the »1 1 1 1«, which can be full of MIDI data (old pattern's notes switching off, new pattern's notes starting, Program Change events, etc), there may be perceptible delays in the transmission of the following Note Ons.

By making each pattern's changeover point a little earlier (»UPBEAT« on, say, »0 4 4 40«, which is 9/768ths), the previous pattern's notes will be switched off a little earlier. So a few pulses later, the MIDI output is free again for the new pattern's notes.

Upbeat summary: so long as an upbeat is less than about twenty pulses, the fact that the previous pattern stops early is imperceptible. The larger the upbeat value, the more this »early stop« becomes noticeable. Obviously, this depends on the tempo.

»CUT«

The changeover point between two patterns can also be moved later than the »1 1 1 1«, in which case you will be cutting into the current pattern and starting, not on its »1 1 1 1« as usual, but on a later bar.

Example: suppose you have an 8-bar pattern, and for the sake of variations, the second time it appears in the list you wish to start it, not on its »1 1 1 1« as usual but on its »5 1 1 1«, ie halfway through.

By increasing that entry's »UPBEAT /CUT« parameter to »5 1 1 1«, this becomes the new start time of the pattern. The entry is now four bars long.

When you »cut« an entry, a »+« appears to the right of its start-bar, and the previous pattern in the chain is lengthened by a corresponding amount.

Note: in many cases, rather than using the same pattern and using »CUT«, it is perhaps simpler to make a separate pattern out of it by copying it, cutting off the unwanted bars using »Cut and Move Pattern«, and inserting it in the chain as normal.

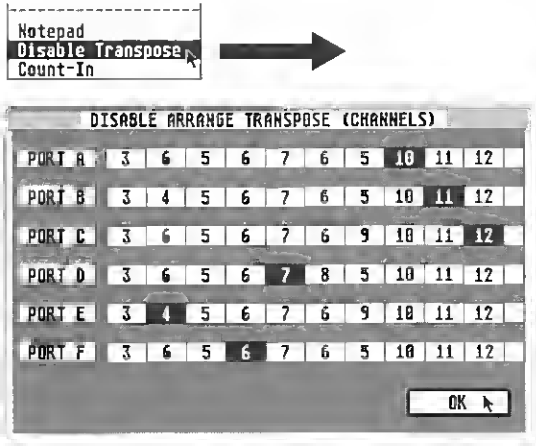
6.3 »Pattern TRANSPOSE«

With the »TRANSPOSE« parameter, the same pattern can be variously transposed throughout the arrange list.

If, for example, the third repeat chorus is to be transposed a semitone up, cursor that entry and enter the value »1« in the »TRANSPOSE« box. That pattern will now be transposed, but only in that entry.

The »Pattern TRANSPOSE« value is added onto any transpose value you may already have in the »TRANSPOSE« track parameter.

6.3.1 »Disable transpose«



When a pattern is transposed, all its tracks are transposed including any drum tracks. This would cause problems of changed instruments where drum machines or samplers worked on a »one drum sound per note« basis.

»Disable Transpose« in the »Options« menu allows you to disallow the arrange transposition on a per-Channel basis. In the window you see six ports each with sixteen Channels; click the Channels of the instruments that should not be transposed (black = disabled).

6.4 »Pattern LENGTH«

Altering an entry's »Pattern LENGTH« value (below the arrange list) determines the length of the current entry and moves all entries that follow it in the same chain.

Always alter entries' lengths with this parameter, never with each entry's start-bar.

»Pattern LENGTH« is a »playback« parameter that does not affect data: a pattern might actually have twenty bars in one of its tracks, but if the entry's length is »4 0 0 0«, only the first four will be heard. Vice versa, too: if a pattern contains four bars of music but

its entry is given a »length« value »20 0 0 0«, you will have sixteen bars› silence after the first four bars!

»Pattern LENGTH« will read »No Limit« if the entry is the last one in the chain (this is because there is no entry below it to stop it). Any music in the pattern could carry on for as long as you liked.

»Pattern LENGTH« will read »ZERO!?« if you have two entries in the same chain that have the same start-bar: click on the »Z« of »ZERO« to produce a bar's value.

Problem: suppose you have finished a song which uses chains a, b and c. The time has come to record it to tape, but you forgot to include a count-in of two bars at the beginning (do not confuse the need for a »song count-in« at the beginning of a song with the automatic Notator/Creator count-in: this automatic count-in is internal and only used to help you record in the sequencer; you will still need a song count-in for the »outside world«, other musicians, vocalists, studio engineers etc).

Solution: the start-bar of the top entry in the list will he »1« (chain not relevant).

– Place the arrange cursor on the top entry.

– Press ›Insert‹ to duplicate the same entry, but in the next chain up.

– Change the new entry's pattern number to a pattern that contains no data.

– Name the entry »count-in« by naming the pattern window.

– Click in chain »a« in the cursor line to make the pattern number move sideways to chain »a«.

– In this »count-in« entry's »Pattern LENGTH« box, add two bars to whatever the current value is: if it is »ZERO!?«, make it »2 0 0 0« (if it were, say, »4 0 0 0« make it »6 0 0 0«, etc).

– Click the »count-in« entry across to chain »b« and repeat the operation of adding two bars to its length.

– Click the »count-in« entry across to chain »c« and repeat the operation of adding two bars to its length.

By adding two bars to the count-in entry at the top of each chain, you have pushed back the whole song by wo bars. It does not matter which chain you leave the count-in bar in. Note how you had to do it to each chain separately since they are independent.

– Lastly, record a count-in in track 1 of the new count-in pattern using a hihat sound (or enter it manually directly into the event editor).

Tip: introducing time signature changes into a song after you have already assembled the arrange list will always result in pattern length changes (producing unexpected start-bar and pattern position values). This is bound to happen (see Chapter 5 »Positioning«, section 1.3 »Time signatures«), but easy to correct: each time you introduce a time signature, correct the lengths of the entries throughout the list.

Note: it makes sense to introduce time signatures changes before assembling the arrange list.

6.5 Arrange mutes

By clicking in the »MUTE« column, any or all of the sixteen tracks of a pattern can be switched off for the duration of that entry in the arrange list: click the desired entry, then click the desired mute boxes (black means muted).

This is not a »Record« function: you should be in »Stop« mode. When you click on an entry in chain »b«, the »MUTES« column will show mutes 17 to 32, in chain »c« it will show 33 to 48, in chain »d« it will show 49 to 64: this helps to emphasize that there are up to 64 tracks available.

The muting and demuting of tracks allows variations of the same pattern if it is entered more than once in the arrange list. If for example you have a synth solo in the repeated chorus pattern which you only want to hear after the singing has finished, mute its track for every entry where there is singing, then leave it open after that.

Do not confuse these arrange mutes with the »screen recording« mutes: see Chapter 7: »Recording«, section 1.2.2 »Screen recording«.

The two types of mute use the same mute column, so they can interact. You could, for example, start an entry with a track muted, then demute it with one of the »screen recording« demutes halfway through the entry: you would only need a demute for that track, because the entry would always start with it muted.

The mutes can be globally enabled or disabled for up to all 32 tracks in a double-pattern, as follows:

DEMUTE ALL left-click the word »MUTE« at the top of the mutes column to demute all the tracks in the pattern. A second left-click restores the previous mutes status.

MUTE ALL right-click the word »MUTE« at the top of the mutes column to mute all the tracks in the pattern. A second right-click restores the previous mutes status.

NB: if the other mouse button is clicked during Demute or Mute All, you lose the previous mutes status.

SOLO MUTE clicking a track's mute while pressing ›Shift‹ will mute all the other tracks leaving the clicked one unmuted. Left-clicking the word MUTE removes all the mutes.

MUTE INVERT clicking the word »MUTE« at the top of the mutes column while pressing ›Alternate‹ inverts the mute status for that pattern (demuted tracks become muted and vice versa). A second click restores the original status.

These four functions Demute/Mute All, Solo and Invert Mute cannot be recorded in realtime using the Screen recording function.

6.5.1 Arrange mute groups

You can store and recall up to ten groups of track mutes and demutes for up to sixteen tracks at a time.

To store – Shift-Alternate-F1 ... F10‹
To recall – ›Alternate-F1 ... F10‹.

These can be used for the arrange mutes, or for »screen recording«, since they can be recorded in realtime.

7. Arrange chain »hide«

A whole chain can be switched off by clicking the letter a, b, c, d at the top of each chain so its background goes black.

This setting cannot be stored and remains active even when you load a new song (beware!).

8. »Arrange mute-, arrange transpose-, arrange upbeat copy«

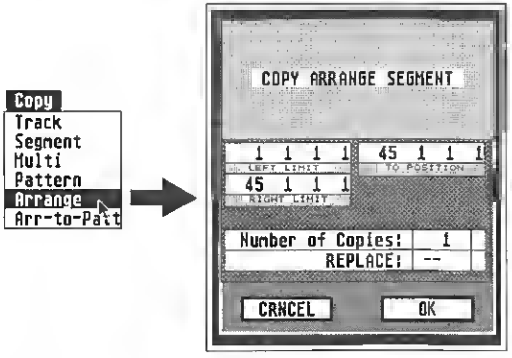
Some pattern parameters of the cursored entry can be copied over to the next entry in the arrange list with these keys:

›T‹ – will copy the Transpose parameter
›B‹ – will copy the Upbeat parameter
›U‹ – will copy the Mute parameters

Pressing ›Shift‹ at the same time as the key will copy the parameter to *all* the following entries in the arrange list.

Example: to transpose the whole arrange list, place the cursor on the top entry, select the »Pattern TRANSPOSE« value and press ›Shift-T‹. To remove them all, place the cursor on the top entry, return the transpose value to zero and press ›Shift-T‹.

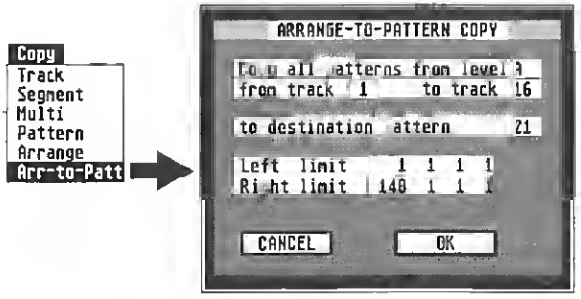
9. »Arrange copy«



You can copy a whole group of entries elsewhere in the arrange list with »Arrange Copy« in the »Copy« menu. It follows the same principle as »Segment Copy« *(Chapter 18, section 3)*.

10. »Arrange-to-pattern copy«

»Arrange-to-Pattern« in the »Copy« menu allows you to make one long pattern out of a sequence of entries in one of the arrange chains.



It works rather like an automatic »Multicopy« function on all sixteen tracks of the patterns in one chain. The notes and other events of all the »source« patterns in one chain are copied one after the other into the corresponding tracks of the destination pattern.

The »LIMIT« positions determine the portion of the arrange list that will be copied: to copy the whole list, ensure the limits are wide enough. You can also determine which tracks will be included in the copying.

The destination tracks must be empty, though you can »initialize« them first with the desired parameters, including track name (Shift-Backspace).

If you do not initialize destination tracks, the first pattern's track parameters will be adopted by the destination pattern.

You should therefore check to see how that will affect your different tracks. If necessary, you must use »Normalize« or »Process Data« to alter the events themselves so they do not need the track parameters. If any tracks use the »LOOP« track parameter, use »Segment Copy« to alter the looped segments into actual data.

The arrange mode's pattern parameters are also ignored, with the exception of »Pattern LENGTH«.

If any entries have a »Pattern TRANSPOSE« value or any other pattern parameter value, you will need to apply those to the tracks first, then »Normalize« etc them. With arrange mutes (or »screen recording« mutes), the relevant data will have to be deleted before using »Arrange-to-pattern Copy«. The exception is the »Pattern LENGTH« value: this will determine the actual length of the pattern that is copied over, so if a length value is »4 0 0 0« but there are twelve bars in the pattern, only the first four will be copied.

The assignment of MIDI Channels to tracks will affect the result.

You must make sure that the same track numbers contain the same MIDI Channels throughout the arrange list before you copy (eg all track 1's = Channel 4, all track 2's = Channel 1 etc).

The program will check the track-to-Channel allocation, and if it detects any mistake, the message »Tracks with different Channels will be accepted /ignored« will appear, giving you the chance to include or reject tracks whose Channels are different from the first pattern's Channels.

Uses for »Arrange-to-pattern Copy«: the two main uses are converting an arrange chain into a pattern prior to printing (Notator's printing capability is based on the contents of one pattern), and for preparing a »Standard MIDI File« (see Chapter 28 »Data Management«, section 6).

11. Deleting the arrange list

»New Arrange« in the »File« menu deletes the whole arrange list, without first issuing a warning: using »UNDO« straight afterward restores the list if you made a mistake.

12. Recording in the arrange mode

Recording is possible with the arrange mode switched on; there are no limits to what you can do.

In fact you often need to have the arrange mode switched on to be able to hear what is happening in the other chains while you record.

When you are in record mode, the program keeps you in the current track, whatever the song is doing in the arrange list: it can be switching entries as the song progresses (you will see the arrange cursor move up the entries in the list), but you are quite safe: you will stay in your record track. The recording does *not* jump from entry to entry with the arrange cursor.

When you go into record mode with the arrange mode on, the count-in means the program will start one bar back (this depends on the count-in) so that you will hear what is in the song as well as the normal metronome »click«.

It is important to ensure you are in the right *track* before entering record mode: so long as that is correct when you enter record mode, then even if the arrange cursor is in the wrong place, no harm will come of it.

In the event list, the time positions of recorded notes and other events are always related to the »1 1 1 1« of the current pattern, not to the »1 1 1 1« of the arrange list.

If you record a track in a pattern which has been entered into the arrange list at various points, the piece which you have recorded will, of course, also play at those various points.

If you find, once you have recorded a track, that the notes are not where you expect them to be, use the event list's »Insert Mode« to move them.

The length of a pattern is as long as its longest track. In practice, though, the length is determined by the »Pattern LENGTH« value: so even if you recorded twenty bars of solo while the song was playing, if the entry's »Pattern LENGTH« value is »8 0 0 0«, the arrange list will play back the first eight bars only.

Simple recording:

You can approach the recording of a song in one of two ways. We will assume that a pattern represents a song section (verse, chorus etc):

1. Record each pattern separately then assemble them afterwards in the arrange list. This is a rather inflexible way to work and can produce a rather disjointed result. You run into problems when you need more simultaneous tracks.

or

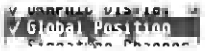
2. Assemble a chain of entries containing as-yet empty patterns. Enter the desired pattern numbers and lengths (if known). You now have a »dummy song« ready to record notes which you may like to save as part of your »AUTOLOAD.SON« *(see Chapter 28, section 2.3)* for future work.

Now place the arrange cursor on the desired entry, place the track cursor on the desired track and click »RECORD«; you hear the last bar of the previous entry (if it contains any music) as count-in (as well as the metronome), then start recording the music. If you are in the middle of recording an 8-bar bassline, and you previously set the pattern's length to »4 0 0 0« in the »Pattern LENGTH« box, don't worry: the arrange list will scroll to the next entry but you can continue recording the bassline. When you have finished, you will need to increase the length to »8 0 0 0« to be able to hear it all.

Remember: all the pattern parameters such as »Pattern LENGTH« are »playback« parameters that can be altered at any time without affecting the data.

13. »Global Positions«

»Global Positions« in the »Edit« menu affects the display of time positions in the event editor and some functions.



With the arrange mode on, when »Global Positions« is enabled, events' time positions will be shown relative to the beginning of the arrange list.

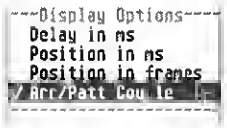
Normally, they are shown relative to the beginning of the pattern.

For »Global Positions« to work properly, ensure the arrange cursor is on the entry whose pattern you wish to inspect.

Example: suppose an event is at »1 1 1 1« in a track. If the pattern is placed in an arrange entry that starts at bar 9, then »Global Positions« will show the event's absolute position in the song: »9 1 1 1«.

The »limit positions« that belong to event-altering functions (»Process Data«, »Transform« etc) and the copy functions (»Segment Copy« etc) will be relative to the arrange list's beginning if »Global Positions« is enabled and the arrange mode is on.

14. »Arrange/pattern couple«



Any change in pattern number in the arrange list will automatically display that pattern in the pattern window. This display-coupling is automatically interrupted when recording with the arrange mode on, so that the pattern does not change while you are in the middle of recording.

The arrange cursor will continue moving down the list while in record mode.

Switching off »Arr/Patt Couple« in the »Flags« menu disables this display-coupling. This has no effect on what you hear, only on what is displayed.

It is normally not necessary to disable »Arr/Patt Couple«. It sometimes is:

Problem 1: chain »a« consists of many short entries, chain »b« contains one entry at the top. You wish to experiment with different track velocities in the chain »b« entry, but whenever you click »START«, the arrange cursor moves down the list and updates the pattern window with the chain »a« entries.

Solution: to concentrate on this pattern while the song is playing, disable »Arr/Patt Couple« and manually select the pattern in the pattern window by clicking the pattern window number.

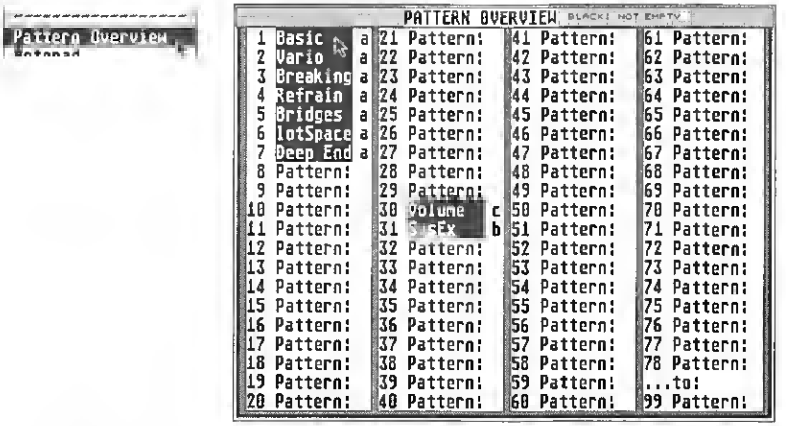
Problem 2: in SMPTE or MIDI Sync modes, the tape position determines the sequencer position. The sequencer position determines the arrange cursor position, which in turn determines which pattern will be displayed.

In a studio, you normally have no control over exactly where the studio engineer starts the tape. But you wish to record in a specific pattern with Creator/Notator synchronized to tape.

Solution: disable »Arr/Patt Couple« and choose the pattern manually (by clicking the pattern window number) so that it does not matter where the tape starts, you are already in the correct pattern and track. All you need to do just before the right point is click »DROP« to enter record mode and start playing.

Compare »Arr/Patt Couple« with »Freeze arrange cursor« (see section 5.2 »CUE« above).

15. »Pattern overview«



The »Pattern overview« window in the »Options« menu displays the status of all 99 patterns at once (not Pattern zero).

For each pattern, it shows:

- The pattern's name, where it has been named at the top of the pattern window.
- Whether it contains recorded tracks (pattern name is shaded).
- Whether it is included in the arrange list, by showing its chain letter.
- Whether it is included in more than one chain, by showing a tick where the chain letter would normally be shown.

To exit from the window, press ‹Return›.

Clicking a pattern exits the window and displays that pattern in the pattern window.

Pressing ‹Shift› while clicking a pattern exits the window and forces that pattern number onto the cursored entry in the arrange list. If the entry is not yet named or has adopted the name of the pattern window *(see section 2 above)*, then the new pattern's name will be forced onto it. If the entry itself has a name via the ‹Shift-Esc› method, that name will continue to be displayed.

15.1 Deleting within Pattern Overview

You may easily delete patterns from within the Pattern Overview window.

Operation:

- Open Pattern Overview by pressing ‹Alternate-O›

– Right-click the pattern that needs deleting.
– The display returns to the main page and a dialog box asks whether you wish to delete that pattern.
– Click OK or press ›Return‹ the contents of the pattern are deleted and you return automatically to Pattern Overview for any more deleting etc.

16. Graphic Arrange Mode (GAM)

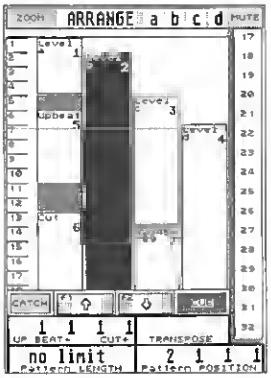
16.1 The basics

The arrangement in the arrange window can also be displayed graphically in the form of beams. Switching is possible between the arrange list and GAM displays without any effect on the musical flow.

The GAM allows an enhanced interaction between user and arrangement, both in the creation and editing of the arrangement, as well as giving a more readable feedback over the structure.

16.2 Activating the GAM

[K] Click the word »ARRANGE« at the top of the arrange window or press ›K‹ to toggle between the arrange list and the GAM.



16.3 Structure

The above display shows how the arrange entries are in the form of vertical beams. Up to four beams can be entered side-by-side, representing chains a to d.

Inside each beam at the top are the name and number of the pattern assigned to that beam.

Any Upbeats or Cuts are shown as grey areas. The grey area within a beam helps distinguish between Upbeats and Cuts and shows to which entry these belong.

Down the lefthand edge of the GAM is the »bar ruler« which shows where the beams are, relative to the song.

The dotted »Song Position Line« (SPL) that moves down the GAM shows the current position of the playback when »Catch« is active.

The »current entry« beam is displayed in reverse video, as is a »selected« beam that you have clicked.

16.4 Zoom

When the GAM is active, the panel to the left of the 

word ARRANGE displays ZOOM: clicking the left/right mouse buttons zooms in/out of the GAM. Keeping a mouse button depressed accelerates the zooming.

NB: as a rule, avoid the extreme zoom factors where you cannot see every bar in the bar ruler.

16.5 Creating GAM entries

Entry beams are created by right-clicking at the desired bar in the desired chain.

NB: if an entry already starts there, you will be warned by a dialog box »Do you really want to duplicate this entry?«. Press cancel: duplicating an entry does not make sense.

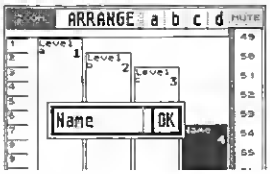
Entries will always be entered at the nearest exact bar. »Pattern POSITION« can then be used should you wish to offset this.

16.6 Deleting GAM entries

An entry beam is deleted by clicking it while keeping ›Shift-Alternate‹ pressed, or by dragging it to the right, out of the arrange window, and releasing it. These methods simply lengthen the previous beam (entry); they keep the following beams in situ.

Alternatively, pressing the ›Delete‹ key deletes the beam and moves all the following beams in that chain earlier to fill the void left by the deleted beam.

16.7 Defining pattern numbers and names

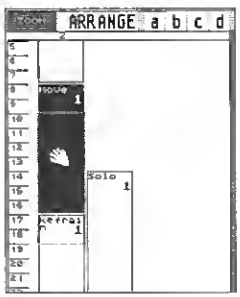


When creating a beam (see above) in the GAM, it takes the pattern number and name of the current pattern in the pattern window.

To change a beam's pattern number, click the little number at the top of the beam.

As in the arrange list, double-clicking the name (click the start of the name) in the beam allows a name different from that of the pattern window to be typed in.

16.8 Moving GAM entries



Dragging a beam, keeping the mouse pointer away from the beam's top right and bottom right corners, produces the familiar little »dragging hand«. Dragging the beam up/down alters its start bar but keeps its end position in situ, so its length will change accordingly. Of course, the length of a preceding beam, if there is one, will alter accordingly as well.

Dragging a beam horizontally moves it to another chain. Its final length will depend on whether any beams are already present in the destination chain: if not, it will acquire a »No Limit« length; if there are, the first beam that follows the new entry will limit the new beam's length.

The length of the entry that preceded the moved beam in the old chain will increase to fill the void left by the moved beam.

NB: very short entries should be dragged by their lefthand top corners. Moving the mouse pointer too far right may change the pattern number or select the length-change feature. If necessary, alter the zoom factor.

Without changing its length, an entry can be moved to a new position if ›Alternate‹ is pressed while dragging it.

This moves all the other entries that follow it to ensure that its length does not change.

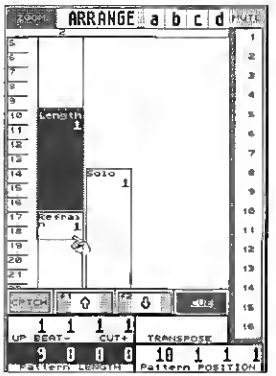
If you move an entry beyond the lower or upper limits of the arrange window, the display will automatically scroll in that direction, allowing the whole operation to be mouse-driven.

16.9 Copying GAM entries

Dragging a beam to a new position while pressing ›Shift‹ copies it.

If you copy an entry beyond the lower or upper limits of the arrange window, the display will automatically scroll in that direction, allowing the whole operation to be mouse-driven.

16.10 Altering the length of GAM entries



Dragging a beam by its lower righthand corner alters its length.

NB: during this operation, the ›Pattern LENGTH‹ panel darkens to confirm you are length-altering.

The effect on the following entries is to move them all earlier/later according to the new length of the beam.

16.11 GAM Upbeat/Cut

Although these parameters must be numerically entered as usual in the Upbeat/Cut panel, they are graphically displayed as grey areas within the beam.

16.12 Moving the GAM display

Click the arrow icons below the arrange window or press ‹F1› or ‹F2› to alter the viewing section of the GAM. Alternatively left-click and hold the CUE icon and move the mouse back and forth.

This will not affect playback and may be done while the sequencer is running.

Tip: in certain situations, such as when dealing with very long arrangements, it may be easier to take advantage of the easy switching between the GAM and arrange list and switch to the arrange list to scroll. Once you have found the desired entry in the list, click it, switch on the GAM, and the selected beam will be ready for you.

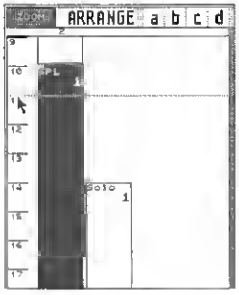
Normally, when you click inside the GAM, the SPL will disappear and the display will freeze to allow you to work on the selected pattern without fear of it scrolling on. However:

Pressing ‹Shift› in combination with the left/right cursor keys will move the GAM cursor from chain to chain, and activate »Catch« at the same time, to allow the SPL to continue scrolling while you are viewing in Play mode.

16.13 Positioning with the SPL

All the usual positioning commands, eg starting from a particular entry, jumping between entries using the bracket keys (which also loads the locators) etc, are available.

Dragging the Song Position Line or clicking in the bar ruler offers more possibilities.



The SPL can be reactivated at any time by using the »Catch« function (press ›L‹).

Drag the SPL at the point where it touches the bar ruler down the lefthand edge of the arrange window, using the left mouse button. This moves the Main Bar Counter bar-by-bar. Alternatively click anywhere in the bar ruler and the SPL will jump there.

NB: dragging the SPL beyond the lower or upper limits of the arrange window automatically makes the GAM scroll in that direction. Additionally pressing the right mouse button when the window limit has been reached speeds up the playback.

In Stop mode, clicking in the bar ruler positions the SPL at the nearest bar: press »Continue« to continue the playback from that point. Clicking in the bar ruler during playback makes the sequence jump to and continue playing from that point.

16.13.1 Locator loading from the GAM

The SPL can be used for easy loading of locators for, say, the Cycle mode.

Left-click in the bar ruler keeping ›Shift‹ pressed to load the left locator. Right-click in the bar ruler keeping ›Shift‹ pressed to load the right locator.

16.13.2 GAM scrub mode

The GAM allows a certain amount of audio »scrubbing« to help, for instance, find a problem in the music.

Operation: Using the right mouse button, click and drag anywhere in the bar ruler, moving up and down. You will hear the MIDI playback in both directions, and the SPL jumps to where the mouse pointer should be.

Example: use this scrub mode as a simple way to find a bad note (ARRANGE is ON, the sequencer is stopped): now right-click and hold (don't move the mouse!) a position just before the bad note. This starts the sequencer.

As soon as you hear the note, release the mouse button.

Enter the event editor of the track with the bad note and click »Catch« or press ›L‹. The cursor should be highlighting the bad note or one very close to it. The search is over!

NB: scrubbing also works when the sequencer is already running. Playback continues from the point where you released the mouse button.

The zoom factor affects the smoothness of the scrubbing.

»TRUE PROGRAM« is deactivated during scrubbing to prevent Program Change commands clogging up the MIDI bus or confusing synthesizers.. To hear Program Changes, use the left mouse button when clicking in the bar ruler.

16.14 Chain-specific functions

The following functions apply to the current chain only when the GAM is active:

Arrange mute Copy – ›U‹ and ›Shift-U‹

Upbeat/Cut Copy – ›B‹ and ›Shift-B‹

Transpose Copy – ›T‹ and ›Shift-T‹

›(‹ and ›)‹ in calculator keypad: jumps to the next entry in the chain and loads the locators.

Tip: to apply the above functions to all the entries whatever chain they are in, switch to the arrange list.

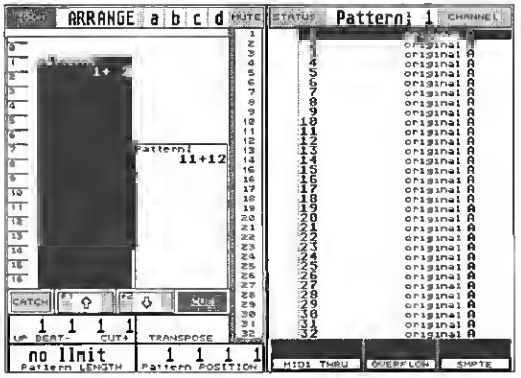
After selecting an entry, the arrange cursor remains on that entry until another entry is selected.

In Play mode when ›Catch‹ is active, only those patterns of the current chain will be displayed in the pattern window.

Use ›Shift‹ with the cursor keys to view other chains: this function includes the ›Catch‹ command so that the SPL continues to scroll.

16.15 Advice re double-pattern entries

In the GAM, entries relating to double-patterns can only be entered in chains a and c. The beams become double-width and the pattern name/number is shown at the top of the beam, including the number of the ›second half‹ of the double-pattern.



The above display shows that patterns 2 and 12 are included within patterns 1 and 11 respectively. The pattern window itself will only show the number of the first pattern.

16.16 Arrange SAVE/LOAD

The arrangement can be separately saved to and loaded from disk from within the »File« menu.

Chapter 21

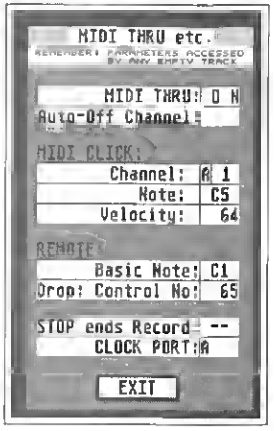
Overall Program Settings

21

1. MIDI Thru

The MIDI Thru function's job is to immediately pass on data that arrives at the MIDI Input to a MIDI Output and on to a MIDI instrument. This makes it possible to simultaneously hear the instrument that you want to record with, without having to wait for playback.

Click »MIDI Thru« (»MIDI« menu) or click the »MIDI THRU« box below the pattern window to open the MIDI Thru window. Its top line switches the MIDI Thru function on/off. Normally, leave it on.



The second line down controls the »Auto-Off Channel« function.

If a Channel 1 to 16 is inserted here, the MIDI Thru function will be switched off for that MIDI Channel only. This solves the problem experienced by synthesizers with no »MIDI Local« function which are being used as master keyboard: without »Auto-Off Channel«, their voices would be triggered twice, once internally by the instrument's keyboard, and again via MIDI, leading to a loss of polyphony and »phased« sounds. For these keyboards, set »Auto-Off Channel« to match their receive Channel.

Tip: switch your master keyboard's »MIDI Local« function off, and switch »Auto-Off Channel« to »blank«, then save that setting with the »AUTOLOAD.SON« (see Chapter 28: »Data Management«, section 2.3).

The track parameters of any *empty* track are used by Creator/Notator's MIDI Thru function, and so affect what you are playing live *(see also Chapter 6 »Tracks«, section 1.2)*.

For example, to re-direct incoming MIDI data to Channel 3 on port A, select »A 3« in an empty track's »CHANNEL« track parameter: the master keyboard's transmit Channel is irrelevant.

Other track parameters (eg »TRANSPOSE«) also affect your live playing *(see Chapter 6 »Tracks«, section 3)*.

A recorded/initialized track is removed from the MIDI Thru function: placing the track cursor on it does not automatically access its track parameters (»CHANNEL« etc):

Place the track cursor on the recorded track and press »#« (USA »\«) to auto-copy its parameters to all the empty tracks, thereby dictating the MIDI Thru function.

The MIDI Thru function is one of Creator/Notator's realtime functions *(see Chapter 23 »Realtime MIDI Functions«)*.

The »MIDI THRU« box below the pattern window shows the velocity of the notes you are playing live in the form of a horizontal beam, if the MIDI Thru function is switched on.

2. MIDI Click in Record and Playback

The metronome during Record and Punch modes (including count-in) can be heard by turning up the Atari monitor's loudspeaker. It is often too quiet for certain situations. It is therefore possible to send the metronome to any MIDI instrument as a short note: the »MIDI Click« *(see also Chapter 7 »Recording«, section 2)*.

The MIDI Click settings are in the »MIDI Thru« window (»MIDI« menu).

You may select the MIDI Out port (A to F), the MIDI Channel within the port, the pitch and the velocity of the note.

The MIDI Click may be switched on/off with »MIDI Click« (»MIDI« menu).

»Play Click« (»Flags« menu) switches the MIDI Click on/off when in Playback mode; it defaults to »off«.

3. Key Remote

The »REMOTE« settings in the »MIDI Thru« window (»MIDI« menu) affect the MIDI »transport remote control« of some Creator/Notator's functions *(see also Chapter 5 »Positioning«, section 4.2 »Transport remote via MIDI«).*

Here, select the »basic note« of the keyboard section being used as the remote controller: this note (default »C1«) is the »START« remote control. The other functions are spread over the following ten semitones.

In addition, the »DROP« function (in and out of Record mode) can be controlled via a footswitch (eg MIDI Controller 65 »Portamento«).

»Key Remote« (»Flags« menu) activates these settings.

4. »STOP ends Record«

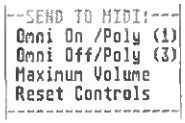
»STOP ends Record« in the »MIDI Thru« window (»MIDI« menu) determines whether a Stop command received while Creator/Notator are in »MIDI Sync« or »SMPTE Sync« modes should end a recording as well.

If »STOP ends Record« is switched off, the tape can be started, stopped and spooled and the sequencer stays in record mode. If the same segment of tape is repeated, existing notes in the sequencer's record track will be heard, and any new notes you play will be merged in. If track parameters »QUANTIZE« or »GROOVE« are being used, time correction will occur whenever and wherever the tape is started. Clicking »STOP« or pressing ‹Enter› will end the recording.

5. MIDI Clock output port

»CLOCK PORT« in the »MIDI Thru« window (»MIDI« menu) determines the MIDI Output port (A to F) through which the MIDI Clock data is sent (only one port at a time).

6. MIDI Mode messages



The »MIDI« menu allows various MIDI messages to be sent to the MIDI devices connected to Creator/Notator. Not all devices understand these messages (refer to their MIDI implementation charts).

Omni On/POLY (1) click this to switch the receiving devices to »Omni On« (MIDI mode 1). In this mode, a device reacts to every Channel it receives. This is usually never used in a normal MIDI system.

Omni Off/POLY (3) click this to switch the receiving devices to »Omni Off« (MIDI mode 3). Here, devices or voices with different Channels react only to data sent on corresponding Channels. This is the standard mode used in a multi-device MIDI system.

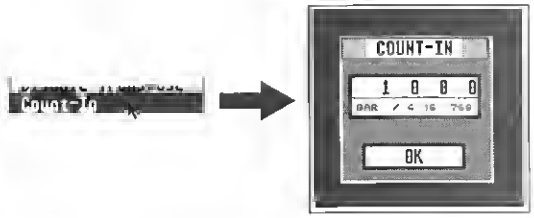
Maximum Volume switches all devices to maximum Volume (Controller 7, value »127«). Use this after, say, an RMG fade out. If ever a device is strangely quiet for no apparant reason, try clicking this option. »Maximum Volume« is not part of the general »Reset Controls« command.

Reset Controls resets the receiving devices' Pitch Bend, Modulation Wheel, Channel Pressure, Sustain footswitch (MIDI Controller 64) etc, and sends an »All Notes Off« message to stop any droning note (it is often quicker to use the »Help« key or to press the »Stop« key (»Enter«) twice).

7. Count-in

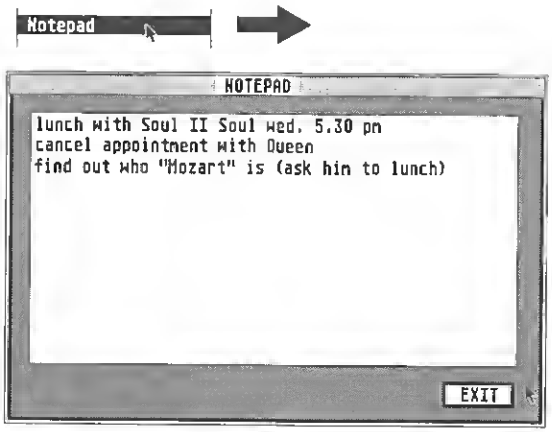
The length of the recording count-in can be altered under »Count-In« (»Options« menu) *(see also Chapter 7 »Recording«, section 2)*. The display is expressed in the usual bar, beat etc form, eg »1 0 0 0« means exactly one bar's count-in. Store the length in the »AUTOLOAD.SON« *(Chapter 28 »Data Management«, section 2.3)*.

You may record during the count-in *(see also Chapter 20 »Arrange Mode«, section 6.2)*.



8. Notepad

Press ›Alternate-N‹, click »Notepad« (»Options« menu) or right-click the »OVERFLOW« box below the pattern window to open the Notepad. The notepad is saved to disk as part of a song, and can be transferred between songs via »Load System« *(Chapter 28 »Data Management«, section 3)*.



Use it to make notes about your song etc. ›Esc‹ deletes a line.

8.1 Auto-open Notepad

Type an exclamation mark ! in the top left corner of the notepad, then save the notepad with a song: this will automatically open the notepad next time the song is loaded.

9. »MIDI Reset on Single Stop«

Keystroke ›Shift-Help‹ allows a single Stop command to send the MIDI system reset commands (normally, a double Stop is needed). Useful in »MIDI« or »SMPTE« Sync modes where some older synthesizers can have droning notes when the tape stops: it saves having always to press the ›Help‹ key.

10. Keystroke: ›Help‹

The ›Help‹ key resets the MIDI instruments and stops droning notes *(see section 6 above)*.

11. RS-232 and »Edit message« reset

Keystroke ›Shift-J‹ resets the RS-232 port at the back of the computer (where you attach Export). It also restores the »Do you want to create a new track?« message if you once clicked the »No message« option when entering an empty track's event editor.

Chapter 22

Realtime MIDI Generator (RMG)

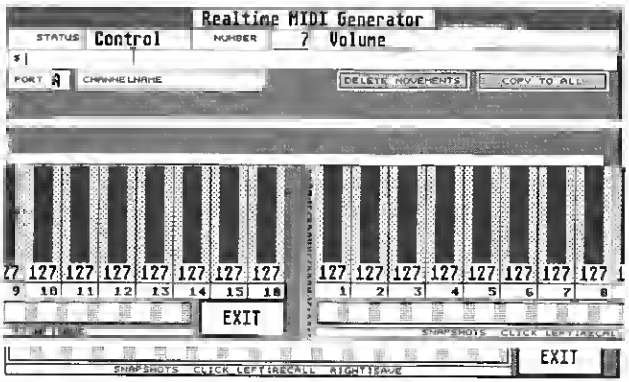
22

1. The basics

The »Realtime MIDI Generator«, called the »RMG« from now on, is a kind of multi-purpose automated MIDI mixer with a large number of different control elements allowing you to generate any type of MIDI data.

Each time you move the »faders«, MIDI events are generated, similar to what happens when you move a synth's modulation wheel.

The RMG is just a tool for generating MIDI events, not an independent sequencer in its own right. In Record mode, the data that is generated by the fader movements is recorded in the current track. In Playback mode, its faders move automatically because the RMG constantly monitors the MIDI output of the sequencer and reacts to certain types of MIDI event.



Note: for reasons of memory conservation, the »moving faders« feature applies only to Control 7, Program and P_USER 5 to 12 events. All other types of event will be recorded and played back as normal, but without moving faders.

To save particular combinations of values for another day, which have been generated by the RMG, data must have been recorded in a track which can then be saved with the current song. MIDI data cannot be saved »in« the RMG.

To save particular fader *positions*, see »Snapshots« *(section 7)*.

The RMG is designed specially to produce flowing, dynamic chains of events of the kind required to change volume or sound characteristics. By using a single multi-timbral synthesizer you could, in theory, work without any other audio components apart from an amplifier and speaker.

Note: the only way to get the data in the track back into the synthesizer, is to Start Creator or Notator so that the track plays the data back.

2. RMG components

Enter the RMG by clicking the »RMG« box (to the left of the »EDIT« box on the main page), or press »M«.

You will see sixteen beam-shaped vertical fader controls. Above each of the faders you will see the first four characters of the MIDI Channel name.

If you click one of the faders, the full MIDI Channel name will appear in the »CHANNELNAME« box (provided you have already named it on the main page).

The fader numbers 1 to 16 correspond to MIDI Channels 1 to 16, *not track numbers!*; for example, fader 3 will transmit on Channel 3.

The MIDI output port is determined in the »PORT« box to the left of »CHANNELNAME«.

To be able to send on all sixteen Channels at once, the MIDI Thru function *(see Chapter 21 »Overall Program Functions«, section 1)* is automatically switched to »original« CHANNEL on entering the RMG. Your master keyboard's transmit Channel will not be re-directed to another Channel.

If you want to be able to play live on a particular instrument's Channel while in the RMG, you must alter your master keyboard's transmit Channel.

3. Generating various types of event

By selecting a status in the top line of the RMG, different types of MIDI event can be generated.

The »STATUS« box allows you to select any MIDI status except for »NOTE«, plus »P_USER« events 5 to 12.

The »P_USER« status allows you to freely define MIDI messages (see Chapter 8 »Types Of Event«).

The »NUMBER« box to the right of »STATUS« determines its »first data byte« *(see Chapters 8 »Types Of Event« and 9 »Editing in the Event Editor«)*. The names of Controller events are displayed with their numbers, as in the event list in the event editor; for polyphonic Aftertouch (Poly Pressure), the note number is shown.

The »second data byte« value, as with Hyper Edit, is determined by moving the fader. The only exceptions are »Program« and »C-Press«, where the fader transmits the »first data byte«.

3.1 Overview of fader assignments

STATUS	»NUMBER«	FADER
Poly Pressure	Note number	Pressure intensity
Control Change	Control Number	Control amount
Program Change	(no meaning)	Program number
Channel Pressure	(no meaning)	Pressure intensity
Pitch Wheel	(no meaning)	Pitch bend amount
P_USER 5 + 9	(user-definable)	definable message
P_USER 6 + 10	(user-definable)	definable message
P_USER 7 + 11	(user-definable)	definable message
P_USER 8 + 12	(user-definable)	definable message

Note: the »Program« status represents a special case. Although all program changes produced by moving the fader during recording are sent to the MIDI device, Creator/Notator will only record the fader's last value (when you release the mouse button). This makes it possible to find the right sounds while hearing them change, without recording the search process itself.

4. Using the faders

The RMG faders can be moved using either mouse button.

If you point at a fader and hold the ›Control‹ key, moving the mouse controls the fader movement without restricting you to moving within the fader's own box. To zero a fader, click either of the mouse buttons while holding ›Control‹.

The numeric value which appears beneath the faders can be increased or decreased by direct clicking with the left/right mouse buttons. This changes the value evenly.

You can also enter specific values using the computer's typewriter keypad number keys (click the fader first). Not using the calculator keypad to do this ensures no start command is given via the large ›0‹ key.

4.1 Recording with the RMG

When you click the Creator/Notator »RECORD« control, you are able to record the MIDI data that is generated by the faders into the current track.

Please note: the data is not recorded in the RMG itself. You are using the RMG as a tool to record data into the sequencer's current track.

If the track is empty, recording takes place as normal, with the added feature that the track is switched to »original« »CHANNEL« on the main page.

If the track already contains data, clicking »RECORD« produces the following dialog box message: »Record will delete old track: RECORD/DUB?«. If you click the »RECORD« box, this completely deletes the current track before starting the recording; if »DUB« is clicked, this keeps the current data and merges the new data in with it.

Use »DUB« after using »DELETE MOVEMENTS« if you wish to correct a Channel's data.

Beware that generating more events on a certain Channel can lead to »fader position conflicts«, if existing data already exists for that Channel: this is why »repairs« to a Channel must be preceded by a deletion *(see »DELETE MOVEMENTS« below)*.

4.2 »DELETE MOVEMENTS«

If the MIDI data relating to the movements of several faders is recorded in one track, and a Channel's data need correcting, it is necessary to first delete the bad data before recording the new.

»DELETE MOVEMENTS« deletes the data of one Channel, either throughout the track, or only while the mouse button is held:

Deleting a Channel throughout a track: click the fader whose data needs deleting, then click the »DELETE MOVEMENTS« icon. A dialog box appears in which you click »DELETE« or press ›Return‹: this deletes all the events in the current track, whose events had been generated by the clicked fader.

Deleting some of the Channel's events: to delete only some of the Channel's events, click the fader, start the sequencer and right-click and hold the »DELETE MOVEMENTS« icon at the desired point: the »busy bee« appears. Release the mouse button when you wish to stop erasing, and a dialog box appears in which you click »DELETE«.

Note: »DELETE MOVEMENTS« deletes data by status and »first data byte«, so that clicking the icon with, say, »Control 7« will only delete volume events from a track that contains for instance a mixture of »CONTROL 7« events and P_USER 5 events.

5. RMG Example: Volume (Control 7)

Suppose you want to vary the volume of a multitimbral expander.

The RMG defaults to status »CONTROL« with »NUMBER« »7« (Control 7 = MIDI Volume). Install this setting if necessary.

- Select an empty track and ensure »CYCLE« is off.
- Click »RECORD« and drag the appropriate Channels' faders up and down. The volumes of the synth sounds should now change as you move the faders.
- Click »START«. All the fader movements have been recorded in the current track and you will be able to watch them move in Playback mode: the current track contains all the Control 7 data.

If you have problems, check the following:

- Have you selected the correct port?
- Did you move the correct faders/MIDI Channels?
- Is the synthesizer able to understand MIDI Volume messages?

Note 1: most drum machines do not support MIDI Control 7.

Note 2: switch on »TRUE Volume« (»MIDI« menu) so that each time you position Creator/Notator, the current MIDI Volume settings are transmitted.

Note 3: the RMG can only be active for one port at a time, so never change ports during recording or the results will be unusable.

6. »COPY TO ALL«

Clicking »COPY TO ALL« copies the value of the current fader (with the flashing box) to all the other faders and transmits the values over MIDI. In Record mode, these values are recorded in the current track.

This function is useful if you want to set all the MIDI devices to full or zero volume.

7. Fader position snapshots

Along the bottom of the RMG screen there are sixteen small boxes. These can show a miniaturized snapshot of the fader positions and are used to store and recall the positions of all sixteen faders. If the sequencer is in Record mode, the action of recalling the snapshots injects all the fader positions into the current track simultaneously.

Right-click a snapshot box to store the current fader positions. This gives you a mini-representation of the current positions.

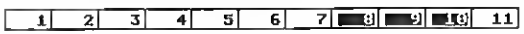
Left-click a snapshot box to immediately recall the stored fader combination and transmit the positions over MIDI.

The snapshot function can be used while the sequencer is running to store and recall positions. It does not store fader movements.

Do not confuse storing and recalling snapshots with recording and playing back MIDI data. Storing and recalling MIDI data is done by recording data in a track, then playing it back. The snapshot function stores fader positions within the RMG only: the events generated by left-clicking a snapshot are not stored in the RMG, but in the current track if in Record mode. Recording events in a track (by moving faders or clicking snapshots) and saving the song to disk is the only way to save your RMG results.

8. Fader grouping

Immediately below the numerical display of the current fader positions there are sixteen small boxes showing the fader numbers (1 to 16).



To group faders, left-click the boxes (black) of the faders you require. Moving one of the faders in this group moves the whole group, keeping the faders' relative positions.

Ideas: by assigning faders to a group, you can fade out all the instruments on one port, or maybe just those »related« to each other, such as drums, percussion and bass.

Faders can be moved separately, whether or not they belong to a group, by using the right mouse button.

Note 1: the relative differences between faders in a fader group will he kept as long as the mimimum and maximum values are not reached. If a group is faded as far down as possible, all the faders will be given value »0« and will then be equal.

Note 2: bear in mind that larger groups can produce enormous quantities of data which very quickly fill up the memory, especially if you have a one Megabyte computer. Try to be economical in your use of fade-ins, fade-outs and group movements in Record mode.

9. Fader grouping snapshots

9.1 Storing and recalling fader group snapshots

You can store up to sixteen lots of faders groupings in the snapshot boxes as follows:

- Form a group as required
- Store the group by holding down the ›Shift‹ key while right-clicking one of the snapshot boxes.

To recall a group, hold down the ›Shift‹ key while left-clicking the snapshot box.

No MIDI data will be transmitted while you are doing this. Thegroup you have selected is now active and can be controlled collectively.

9.2 Storing and recalling some (not all) faders positions

It is not always desirahle for every fader's data to be transmitted over MIDI when a snapshot is recalled. The RMG can therefore restrict the storing and recalling of fader positions to selected faders.

Storing:

- Form a group by clicking the group boxes.
- Set the desired fader positions (using the right mouse button).
- Store this setting by holding down the ›Shift‹ key and right-clicking a snapshot box.

Recalling:

- Left-click the desired snapshot box *without* holding ›Shift‹.

This recalls the fader positions of the group only and transmits them via MIDI. The faders not in this group will not be affected and will not transmit their data via MIDI.

Note: RMG snapshots are stored with a song and can be loaded into other songs separately using »Load System« (see Chapter 28: »Data Management«, section 3 »Load System«).

10. RMG tips

Snapshots can be used to give each pattern in the arrange list a specific mix volume ratio. To do this, record the snapshot's MIDI data on or before the first beat of the pattern. The easiest way to do this is to select an empty track, select the position you want in the Main Bar Counter with Creator/Notator in Stop mode, click »DROP« to go into Record mode and then left-click the desired snapshot.

If recording a separate mix in a pattern, it is advisable to ensure »TRUE Volume« is on (»MIDI« menu), record the mix a few pulses before the pattern's first beat (eg »_ 4 4 40«), then give the relevant arrange entry its own »UPBEAT« value by the same amount (if they occur before a pattern's first beat, TRUE functions will only be »seen« in the arrange list if there is a corresponding »UPBEAT«). *See also Chapter 20 »Arrange Mode«.*

It is preferable to do the mixing in a separate »No Limit« pattern in its separate arrange chain (eg »d«).

11. User-defined RMG elements (STATUS: P_USER 5 – 12)

11.1 The basics

Each control element (faders, switches or »digital readouts«) is allocated to its own P_USER event: P_USER 5 to 8 have the faders, and P_USER 9 to 12 the switches and digital readouts.

P_USER events 5 to 12 can be used as »stand-ins« for any freely-definable MIDI message *(see also Chapter 8 »Types Of Event«, section 7 »Pseudo events«)*. The data chain allocated to a P_USER event can be up to fourteen individual data bytes long.

If such a P_USER event is generated, it carries out the task set by the definition: it issues a MIDI message based on the data chain allocated to it.

P_USER events themselves can be viewed in the event editor after a recording. The »CHANNEL« column in the event list shows the fader number (1 to 16). The »second data byte« shows the fader value.

The RMG shows a maximum of 32 control elements at a time (sixteen faders plus sixteen switches or digital readouts). Switch between »Sets« by selecting »P_USER 5« to »P_USER 8« in the »STATUS« box. The faders are allocated to P_USER events 5 to 8.

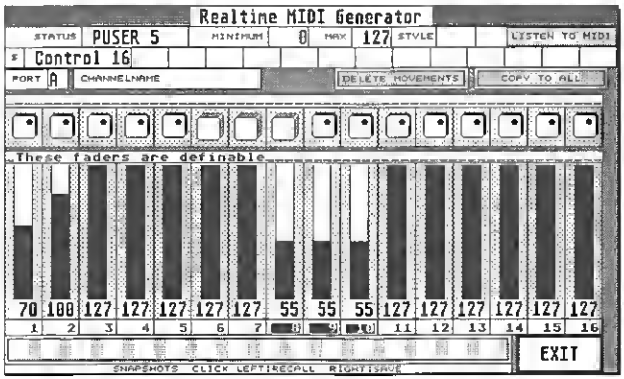
The sixteen switches and digital readout settings are allocated to P_USER events 9 to 12. They do not appear directly in the »STATUS« box, but are shown in the event list after a successful recording and can be edited.

For example, if you select »P_USER 5«, P_USER 9 will receive the switches and digital readouts. P_USER 6 is linked to P_USER 10, P_USER 7 to P_USER 11, P_USER 8 to P_USER 12.

There are therefore 128 freely-definable control elements in the form of faders, switches or digital readouts (4 P_USERs times 32 elements).

11.2 Overview

If you select P_USER 5, 6, 7 or 8 in the RMG's »STATUS« box, the following things will happen on the screen:



The »MINIMUM«, »MAX« AND »STYLE« parameters and the »LISTEN TO MIDI« icon will appear on the top line.

»MINIMUM« and »MAX« are used to set the maximum and minimum values of a control element, whilst »STYLE« is used to choose between the digital readouts and switch elements.

»LISTEN TO MIDI« automatically defines the message chain by analyzing external incoming MIDI data.

The line underneath the »STATUS« box is divided into fourteen little boxes: this is the »definitions line«. The first, slightly larger box shows the first value as a MIDI Status for easy legibility. The other boxes receive the additional values of the definable message.

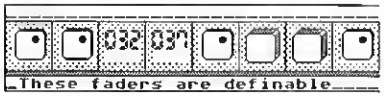
Note: if you want to enter SysEx messages by hand, leave out F 0 (which can be selected in the first box of the »definitions line«) and F 7 (EOX = End Of Exclusive; which is automatically sent by the program).

To switch between decimal, hexadecimal and ASCII displays, click the little »dollar« box (»$«).

In the narrow line below the »COPY TO ALL« icon but above the faders, the switches and digital readouts can be labelled by typing in text.

Directly below this are the sixteen control elements: switches or controllers in the form of three-digit digital readouts.

Below this in turn is a narrow line for entering text which labels the faders.



11.3 Defining the digital readouts, switches and faders

Decide what the top sixteen control elements are to look like, as follows:

- Click one of the control elements with the mouse. Its border will now blink on and off.
- Click the box next to »STYLE« in the top line to choose between a switch or digital readout.
- Now define its lowest and highest values (or the »off« and »on« values for switches) in the »MINIMUM« and »MAX« boxes. For most controllers, 0 will be »off« and 127 will be »on«.

If you have selected the »switch« as controller, the »switch off« setting will correspond to the »MINIMUM« value and the »switch on« setting to the »MAXIMUM«.

- Define any MIDI message in the line below with the fourteen boxes. If you leave a box empty, its value will not be sent. »VAL« means »value« and means that the current fader or switch setting will be sent at that position.

If you have chosen a »continuous« controller (fader, digital readout), you can set and send any value between the minimum and maximum. If parameter changes using SysEx data are involved, an upper and lower limit can be especially useful.

The »MINIMUM« and »MAX« parameters can be defined for all controllers, even faders. The range of a fader's movement on screen is correspondingly limited.

11.4 Example: user-defined faders with Control 10

Suppose you want to allocate the message »Send MIDI Control 10 on Channel 1 with adjustable range« to fader 1. On compatible devices, MIDI Control 10 controls the stereo panning of a sound.

– Choose »STATUS« »P_USER 5« and click fader 1.

– Set the value »Control 1« in the first box of the definitions line. This »1« is the MIDI Channel.

– Enter the MIDI Control number »10« in the next box of the definitions line.

– Enter »VAL« in the next box. This means that this value can be controlled with the fader. In this case it is the »second data byte« of the MIDI Control 10 event (the pan position) which is controlled.

– Now control the synthesizer by moving the fader. Set the master keyboard to transmit on Channel 1 if you want to hear the effect live; set the receiving synth to receive on Channel 1, and the sound you have selected should react to the panning events being generated.

– Click an empty track.

– Record a few fader movements. These will move during Playback.

– Look in the event list to see the events: only the P_USER event has been saved, not the actual panning data.

Note: you can even subsequently alter the »second data byte« of an RMG P_USER event in the event list. However, changing the »first data byte« or fader number will make a nonsense of the allocation you have already set, so this is normally not a good idea.

11.5 »LISTEN TO MIDI«

Manually entering large numbers of long message chains can take a great deal of time and effort. In many cases you can save yourself the trouble by using the »LISTEN« function.

If you click the »LISTEN TO MIDI« box, Creator/Notator will go into a kind of »Record« mode and compare a series of successive incoming MIDI messages. It analyzes which MIDI bytes show a constant value and which show variable values. This means you can allocate a synthesizer function to a fader, switch etc, without entering numbers manually.

Example: suppose you want to automatically allocate the event »Modulation wheel on Channel 1 with variable amount« to a fader using »LISTEN TO MIDI«:

– Click fader 2.

– Click »LISTEN«. The »busy bee« means that the program is waiting for a MIDI message.

– Now *slowly* turn the modulation wheel. The entry should look like this:

STATUS column – P_USER 5
Box 1 – Control 1 (Status »Control«, MIDI-Channel 1)
Box 2 – 1 (Control number, modulation wheel«)
Box 3 – VAL (amount depends on fader)

The modulation data was analyzed as follows:

Message	Status	Channel	1st data byte	2nd data byte
1.	Control	1	1	xx (Value 1)
2.	Control	1	1	yy (Value 2)
Result:	Control	1	1	vv (VAL)

This is just an example of the underlying principle: Mod Wheel data is, of course, easier to produce by just moving the wheel.

Try experimenting with System Exclusive data as well. This method allows the control of synthesizer sound parameters in realtime, which opens up a huge range of possibilities.

Check the following if you encounter any problems:

– Ensure nothing is being filtered in the »Input Handling« window (»MIDI« menu) (no grey boxes).
– The synthesizer etc must be able to actively send the type of data you want.
– If you are using SysEx data, the MIDI device must not send any »Checksums«.
– No »handshaking« communication must be required.
– The length of a single message must not exceed the total length of the message chain (maximum fourteen bytes).

11.5 Example: realtime control over reverb

The following section describes how to control the reverb effect of a Lexicon LXP-1 digital reverb processor in realtime. You can use SysEx data to control various features such as reverb length, pre-delay etc.

Below is a list of SysEx messages, what they mean and a suggested layout for the RMG control elements. This example assumes that the LXP1's receive Channel is set to »16«. If you want to use another Channel, alter the third data byte in the definitions line (here: 47 = Channel 16) to a different value (eg 32 = Channel 1).

Use the first »RMG set« with Status P_USER 5 (and P_USER 9).

Switch 1 (P_USER 9 [Channel 1])

Definition line: | SysExcl | 6 | 2 | 47 | 65 | 0 | VAL | 0 |
(Effect Type)

Style: Digital readout, value range: 0 to 15
This selects the type of effect. Choose »0«, a reverb program.

Switch 2 (P-USER 9 [Channel 2])

Definition line: | SysExcl | 6 | 2 | 47 | 2 | 2 | 0 | VAL |
(Volume)

Style: Switch, value range: 0 to 64.
This switch switches the effect on/off.

Fader 1 (P_USER 5 [Channel 1])

Definition line: | SysExcl | 6 | 2 | 47 | 65 | 0 | VAL | 0 |
(Volume)

Value range: 0 to 127
This fader smoothly controls the amount of effect.

Fader 2 (P_USER 5 [Channel 2])

Definition line: | SysExcl | 6 | 2 | 47 | 0 | 2 | 0 | VAL |
(Decay)

Value range: 0 to 64
This fader controls the reverb length.

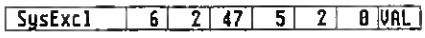
Fader 3 (P_USER 5 [Channel 3])

Definition line: | SysExcl | 6 | 2 | 47 | 1 | 2 | 0 | VAL |
(Decay)

Value range: 0 to 64
This fader controls the reverb effect's pre-delay and early reflections times.

Fader 4 (P_USER 5 [Channel 4])

Definition line:

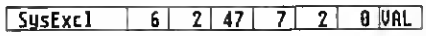


(Size)

Value range: 0 to 64
This fader determines room-size simulation and correlates to the reverb length (fader 2).

Fader 5 (P_USER 5 [Channel 5])

Definition line:

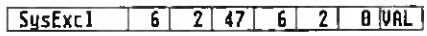


(Diffusion)

Value range: 0 to 64
This fader softens the attack of percussive sounds.

Fader 6 (P_USER 5 [Channel 6])

Definition line:

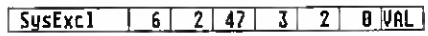


(Feedback)

Value range: 0 to 64 This fader determines the reverb feedback level. Caution: self-oscillating feedback is easily created.

Fader 7 (P_USER 5 [Channel 7])

Definition line:

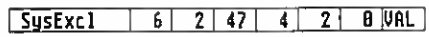


(Low Frequency Amount)

Value range: 0 to 64
This fader controls the amount of low frequencies in the effect (32 = neutral).

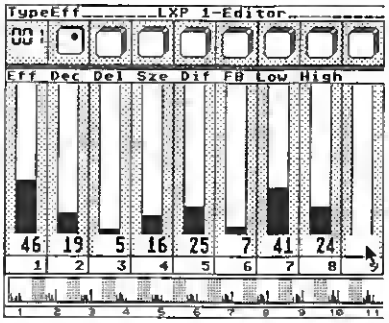
Fader 8 (P_USER 5 [Channel 8])

Definition line:



(High Frequency Amount)

Value range: 0 to 64
This fader controls the amount of high frequencies in the effect (32 = neutral).



These parameters listed above must be entered by hand because the LXP-1 does not send all its data (»Parameter Change«) actively.

Chapter 23

Realtime MIDI Functions

23

Creator and Notator can process MIDI data in a multitude of ways as it flows in, »live«, as you play. Another word for »live« processing is »realtime«: calculations happen so quickly that there is no perceptible delay between the data arriving, being processed, and leaving again.

These realtime functions are comparable to what MIDI processors or MIDI mother keyboards do.

1. Signal flow

When MIDI arrives at one of the MIDI Inputs, it is routed to the following functions in the specified order without perceptible delay:

1. MIDI In
2. Input filter
3. »Realtime Transform« (two stages)
4. MIDI merge of all three Inputs
5. Recording in the current track
6. MIDI Thru function
7. Realtime Ghost output.

2. MIDI Inputs

The Atari's own MIDI In is identified in Creator/Notator by a Roman 1: »I«.

With Unitor, you also have additional Inputs »II« and »III«.

All three Inputs can be received (»merged«) at the same time, even where there is lots of data.

System Exclusive messages (eg sound data) can be received at any of the Inputs. However, because of the non-standard structure of SysEx, merging is not always possible if long SysEx messages are received simultaneously at more than one port. Try to avoid this situation.

If you are not using all three Inputs, use the extremely reliable Inputs II and III in Unitor, especially in live and in SMPTE synchronization situations.

3. Input handling

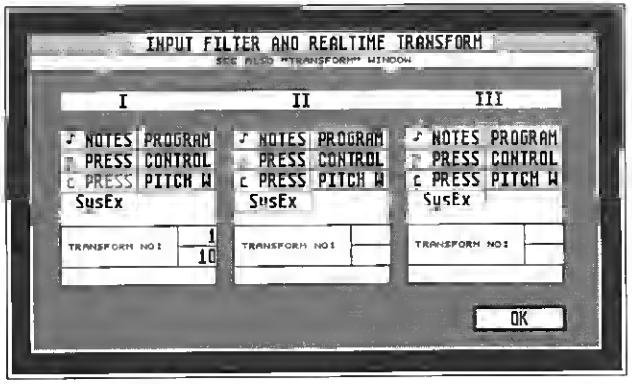
Each Input has its own Input filter and two »REALTIME Transform« stages. »REALTIME Transform« is perhaps the most powerful of the realtime functions, allowing live events to be:

- transformed into virtually anything (eg Pitch wheel into tempo, or inverting the keyboard),
- deleted (eg deleting (filtering out) all incoming Channels 10 to 16 events),
- »transcopied« (eg instantly producing notes an octave above as well).

With REALTIME Transform, your master keyboard vastly increases its power *(see Chapter 24 »Transform«)*.

3.1 Input filter

In the »Input Handling« window (»MIDI« menu), each Input can independently filter out one or more statuses of incoming MIDI event, preventing selected (grey letters-on-white background) events from being recorded.



There is no reason, on grounds of timing or memory conservation, why you should need to filter out Channel or Poly-Pressure (»Aftertouch«) events, because Creator has special ways of dealing with this sort of data such as »Play Algorithm« and »Data Reduction« *(see »Appendix«)*, though if you never make use of Aftertouch data, it would make sense not to record it.

MIDI data can be sent to the »Transform« function in realtime as it arrives at the MIDI Inputs: enter the number of the desired »Transform Sets« (up to two) in each of the Input boxes: *see Chapter 24 »Transform«*.

4. Merging the three Inputs into one track

The MIDI data streams from all three MIDI Inputs are first filtered in the Input filter and sent to the »Transform« function before being mixed together and recorded in one track.

5. MIDI Thru: empty tracks' »track parameters«

If the MIDI Thru function is on *(see Chapter 21 »Overall Program Functions«, section 1 »MIDI Thru«)*, the mixed data is output via ports A–F. The output can be affected by the parameters of any empty track, including realtime transposing, velocity compressing, zoning etc *(see Chapter 6 »Tracks«)*. The Output Filter of an empty track allows events to be filtered out.

6. Realtime MIDI processing

6.1 Realtime ghost tracks

In addition to the MIDI Thru function, you can bring in a further 64 tracks into Creator's realtime MIDI processing capability. These further tracks could be described as »realtime ghost tracks« of the original track. It allows a vast number of realtime possibilities: multi-echo effects, multi-splits, multi-layering, sending to several ports simultaneously *(see Chapter 6 »Tracks«, section 3.11)*, etc.

To make a ghost of another track, enter the master track's pattern and track in the ghost track's »GHOST OF« track parameter.

Any tracks in an active pattern which are »ghosting« an empty track (and therefore the MIDI Thru track), will re-transmit incoming MIDI data in realtime, configured by their own track parameters *(see Chapter 6 »Tracks«, section 3.9 »Ghost of«)*.

Track parameters CHANNEL, TRANSPOSE, VELOCITY, COMPRESSION, DELAY, HIGHEST and LOWEST are possible in a ghost track. By using different HIGHEST/LOWEST parameters in several of the Ghost tracks, different splits can be given different treatments. When trying octave doubling or echo effects on the same MIDI Channel, use each track's output filter to filter out over-layered pitch-bending or modulation.

Echo effects with the DELAY parameter are available only when the sequencer is running. Negative DELAY values, QUANTIZE and GROOVE are, as you might expect, not available in realtime.

Creator/Notator can play music and simultaneously process realtime ghost tracks: the tracks are freely divided between normal recorded tracks, ghost tracks of recorded tracks and realtime ghost tracks of an empty track.

With the arrange mode off, the tracks in the current pattern are at your disposal. When on, there are up to four simultaneous patterns available, depending on the arrange list and the Main Bar Counter position. This means that you have some interesting possibilities with pre-programmed pattern changes containing different variations of realtime ghost tracks.

If you find that your played notes are »swallowed« at the point where there is a pattern change, you reduce the »UPBEAT« value for the current entry at that point.

If you like to work with realtime ghost tracks in the Arrange mode, it may make sense to assign a separate pattern to your ghosts, and assign the pattern to its own arrange chain, eg »d« *(see also Chapter 20 »Arrange Mode«).*

Note: by multiplying the amount of MIDI data being sent from the computer's MIDI Output by using realtime ghost tracks, you may be stretching MIDI's timing capability to its limits, even though the MIDI Input is relatively empty. This might in severe cases lead to delays in the re-transmission of data.

6.2 Examples of realtime ghost tracks

Example 1: sending to two MIDI Channels simultaneously

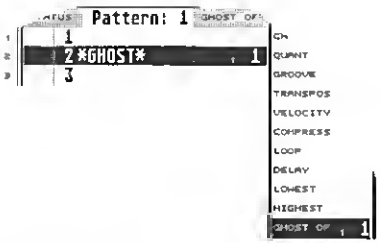
Assign the first Channel in any empty track of the current pattern. This sets the MIDI Thru function's Channel. In a second empty track, enter into its »GHOST OF« track parameter, the pattern and track numbers of the first (empty) track. Give the ghost track a different MIDI Channel.

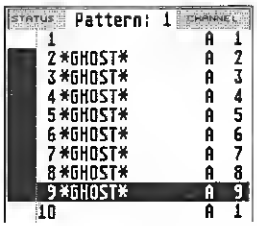
Example 2: sending on up to 96 Channels simultaneously

– Select an empty pattern (pattern 1).

– Leaving track 1 empty, click track 2 and enter »1 1« in its »GHOST OF« parameter.

– For as many instruments/Channels as your MIDI system supports, enter »GHOST OF 1 1« in more tracks. For more than sixteen Channels, continue in the next pattern, not forgetting to enter the patterns in the arrange list in simultaneous chains.





– In each track, select a different MIDI Channel/port.

– Play a note.

Each note you play is re-directed in realtime to all the instruments with the aid of the ghost tracks.

Each track can also receive other parameters as well, such as transposition etc.

Chapter 24

Transform

24

1. The basics

The TRANSFORM function allows the transformation of any type of MIDI and P_USER event. Comprehensive »conditions« determine which events should be transformed and how they should be transformed.

»Transformation« in this sense means »changing into something different« or »conversion«.

TRANSFORM does not generate events (like, say, the RMG). It can only work with events that already exist.

The TRANSFORM window is also used for deleting events that require more than simple deleting *(see section 2.4.1 »DELETE mode«)*.

For simple deleting, please *see Chapter 18 »Copy, Merge, ...«, section 11.4 »Summary of delete functions«*, especially the »Fast Delete/Keep« function.

Most of the »event-altering« functions in the program can be done by TRANSFORM as well, eg transposition, velocity changing etc.

Although carrying out these simpler operations is a more complex procedure in TRANSFORM than using the functions designed for them, all sorts of possibilities open up to carry out even very elaborate and unusual calculations on data.

TRANSFORM, in a way, is on a level above all the other functions. Instead of simply altering a value to another value, the user can combine a number of different alterations within one command, determining the ways and means and strategy involved.

The transformation of MIDI data can take place in realtime before any recording allowing you to hear the change as it happens whether the sequencer is stopped or running.

The range of possibilities is almost unlimited: in realtime, your master keyboard can have user-defined velocity curves, control of volume from the modulation wheel, sustain on/off switching from a key etc.

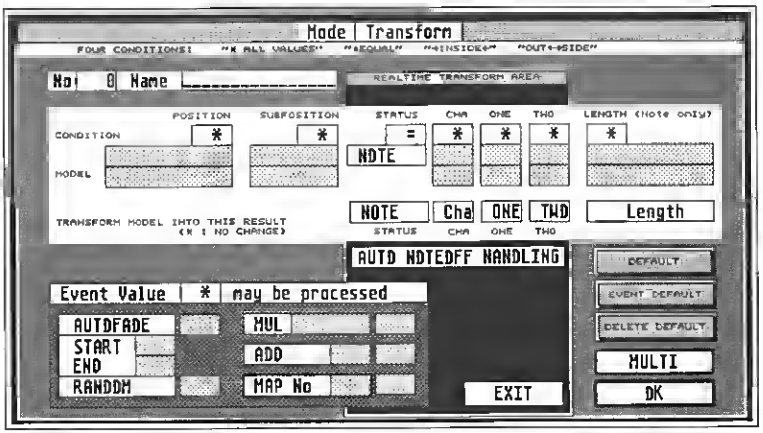
MIDI data can even be transformed into P_USER events *(see Chapter 8 »Types Of Event«)*. This allows commands to be given to the program in realtime from the keyboard, eg controlling track muting, tempo etc from foot pedals, switches, keys, wheels etc.

Transformation can be global (»Multi TRANSFORM«: all patterns, all tracks) as well as local (one pattern, track, track segment), or specific to a position (eg every second beat of the bar).

Positions and note lengths are transformable, and periodic and random transformation of values is possible.

To get the most out of TRANSFORM, a basic knowledge of the MIDI data format is assumed. *(Please read Chapters 8 »Types Of Event« and 9 »Editing in the Event Editor«.)*

2. Overview: functions and defaults



2.1 TRANSFORM and REALTIME TRANSFORM

TRANSFORM allows two different ways of working:

1. Transformation of data already in the computer's memory (RAM): we will call this »TRANSFORM« from now on.
2. Transformation of data in realtime: we will call this »REALTIME TRANSFORM« from now on, because data is transformed just before it is recorded and filed in the RAM. Data is »REALTIME TRANSFORMED« first before passing through the MIDI Thru

function to the Outputs and to a track for any recording *(see Chapter 23 »Realtime MIDI Functions«)*.

2.2 TRANSFORM »Sets«

Click »TRANSFORM« (»Functions« menu) or press ›Alternate-T‹ to open the TRANSFORM window.

There are eleven TRANSFORM »Sets« (complete TRANSFORM windows) in which the desired calculations are determined.

Set 0 is not available for REALTIME TRANSFORM. Use it as your day-to-day Set for casual transformations. Sets 1 to 10 are available for TRANSFORM and REALTIME TRANSFORM. Use them to store pre-programmed transformations etc.

Top left of the window is the Set number (»No«) box and name box. Each Set can have a sixteen character name.

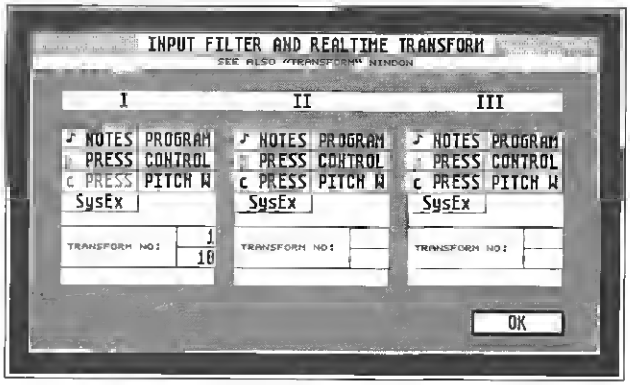
Click the Set number box to select the TRANSFORM Set. 

See section 3 for details of the TRANSFORM Set parameters.

2.3 How a Set can be used for REALTIME TRANSFORM

Each MIDI Input (I, II, III) can be subject to two independent TRANSFORM Sets in realtime. The calculations in the Sets have immediate results in realtime.

Insert the REALTIME TRANSFORM Set number(s) in the »TRANSFORM NO«. boxes of the »Input Handling« window.



Click »Input Handling« (»MIDI« menu) or right-click the »MIDI THRU« box below the pattern window to enter the Input window from the main page.

To enter the Input window from the TRANSFORM window, click the large black area in the lower half of the window: the »REALTIME TRANSFORM AREA«.

2.3.1 Overview: data flow

See also Chapter 23 »Realtime MIDI Functions«.

When REALTIME TRANSFORM is active, MIDI flows as follows:

1. Data arrives at one of the Inputs.
2. The »Input Handling« window can filter events by status, stopping the »REALTIME TRANSFORM Sets from receiving them.
3. The TRANSFORM Sets installed in the Input window are flowed through in turn, ie the results of the top one are sent to the bottom one.
4. After transformation, data flows through the MIDI Thru function and, if necessary, is recorded at the same time.
5. The MIDI Thru function, if switched on, applies the »playback« parameters of the empty tracks to the data flow (CHANNEL, TRANSPOSE, VELOCITY, COMPRESS, LOWEST/HIGHEST and the OUTPUT FILTER) before the data leaves by one of the six Outputs.

Note to 5: set the parameters to zero, and »CHANNEL« to »original« to have the original REALTIME TRANSFORM effect without any MIDI Thru function alteration.

2.4 TRANSFORM modes

There are basically three different TRANSFORM modes: select these in the »Mode« box at the top of the window.

2.4.1 »DELETE« mode

Much deleting can be done more simply using other functions *(see Chapter 18 »Copy, Merge, ...«, section 11.4 for a delete summary)*. However, it is sometimes necessary to set conditions (eg »delete all Channel 3 notes in the first twenty bars« etc): here, the TRANSFORM function is used in its »DELETE« mode.

Clicking »Delete/Keep Events« (»Functions« menu) enters you directly into the TRANSFORM window's »DELETE« mode.

The action of »keeping« is just another way of saying »delete all other events«; therefore »keeping« events is also done in the »DELETE« mode (eg »delete all notes *whose pitches are not* C3«).

See the examples at the end of this Chapter.

Events that match the »model« and its »conditions« are deleted. This is also possible with REALTIME TRANSFORM.

Events which do not match the »Model« or its »Conditions« are untouched.

Note: the »RESULT« line (see section 4.4) and »VALUE PROCESS« box (see section 5) are disabled when the mode is »DELETE«, as you cannot delete events »into« a result.

2.4.2 »TRANSFORM« mode

Events that match the »Model« and its »Conditions« are transformed into the »Result«, with possible further transformation by variables.

Events which do not match the »Model« or its »Conditions« are untouched.

2.4.3 »TransCOPY« mode

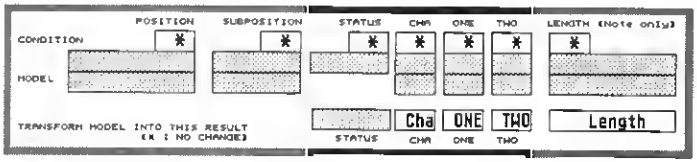
Events that match the »Model« and its »Conditions« are first duplicated. The duplicate is then transformed into the »Result«, with possible further transformation by variables, and inserted back among the original untransformed data.

Events which do not match the »Model« or its »Conditions« are not copied nor transformed.

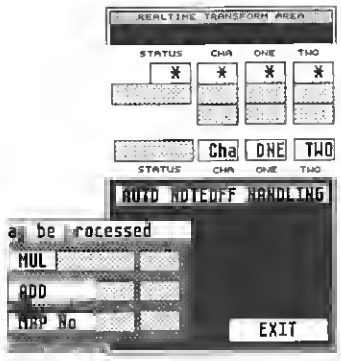
3. Structure of the Transform window

The TRANSFORM window is divided up into several parts:

– The large white »TRANSFORM AREA« in the upper half of the window, where you decide how and how much of the data should be transformed (»MODEL« and »CONDITIONS«) and what the »RESULT« should be.



– The large black »REALTIME TRANSFORM AREA« partly obscured by the TRANSFORM area *(see section 2.3 above)*.



– The large »PROCESS VALUE AREA« below left in the window. If the »RESULT« line *allows*, individual event values can be further modified here by various calculations.

4. »TRANSFORM AREA«

4.1 »CONDITIONS«

The various boxes in the top line called »CONDITION« contain »Conditions«. The values in each column of the »MODEL« line must match these conditions if they are to be transformed.

There are four conditions:

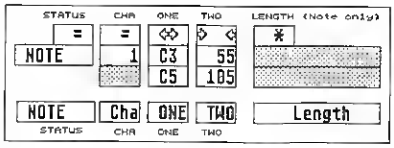
* – means »all values are accepted for transformation, there are no conditions attached to this column«.

= – means »only values that match the specific »MODEL« value in this column will be accepted for transformation«.

>< – means »only values that are within the two »MODEL« values in this column will be accepted for transformation«.

<> – means »only values that are outside the two »MODEL« values in this column will be accepted for transformation«.

4.2 »MODEL«



The second line in the TRANSFORM AREA is the first of the two »MODEL« lines. This contains the value(s) which incoming MIDI events have to match if they are to be transformed.

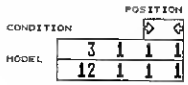
The third line only becomes active when the condition is >< »inside« or <> »outside«, and shows the upper or »right« limit of the condition zone.

In other words: line 2 defines the beginning (»from«) of the zone, line 3 the end (»to«) of the zone.

The »from« and »to« values are always »inclusive/exclusive«, including the »POSITION« columns. *See »Appendix«, section 1.9 »Values: »inclusive/exclusive««.*

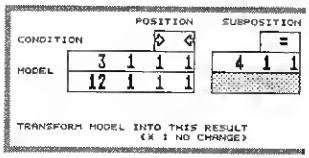
The »MODEL« columns have the following meanings:

POSITION »events in the track that are on this position or are in the segment defined between the two »locators« are accepted for transformation«. Not available in realtime.



Example: transform all notes in bars 3 to 11 inclusive.

SUBPOSITION »within each bar which is within the above segment, events that are on this bar position or between these two bar locators are accepted for transformation«. Not in realtime.



Example: transform all notes in bars 3 to 11 inclusive, but only if they are on the fourth beat of the bar (»4 1 1«).

STATUS »events whose statuses match this status are accepted for transformation«. No »inside/outside« conditions are possible because numerical values are not involved.

Example: only transform events if they are »PROGRAM« events.

CHANNEL »events whose Channels match this or these Channels are accepted for transformation«.

Example: only transform those events whose Channels are 7, 8 or 9.

ONE »events whose »first data bytes« match this or these values are accepted for transformation«.

Example: only transform those events whose pitch is C3.

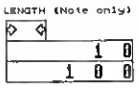
See Chapters 8 »Types Of Event« and 9 »Editing in the Event Editor« for details of the meaning of »first data byte«.

TWO »events whose »second data bytes« match this or these values are accepted for transformation«.

See Chapters 8 »Types Of Event« and 9 »Editing in the Event Editor« for details of the meaning of »second data byte«.

Example: only transform those events whose velocities are from 70 to 126 inclusive.

LENGTH: »notes whose lengths match this or these values are accepted for transformation«.



Example: only transform notes that are 1/16th or longer, but shorter than 1/4.

4.3 »DEFAULT«

The entering of the desired values can also be done from the event editor using the three icons to the right of the window.

DEFAULT in the event list, click an event line and leave the cursor on one of the event lines. Open »TRANSFORM« (eg with ›Alternate-T‹). If you click the »DEFAULT« icon:

- the window as a whole will be restored to its default (zero) setting
- the values of that cursored event will be inserted in the »MODEL« line. These values will become visible when the Conditions are set in the »CONDITIONS« line.

EVENT DEFAULT has the same action as »DEFAULT« with the addition: if you leave the event list cursor in one of the »CHANNEL«, »1«, »2« or »LENGTH« columns, open the TRANSFORM window and click »EVENT DEFAULT«:

- the value you cursored in the event list is installed (visibly) in the corresponding column of the »RESULT« line.

This is the same as using the »Fast TRANSFORM« function, but using the »TRANSFORM« window, therefore allowing other conditions to be added such as limiting the number of bars etc *(see Chapter 9 »Editing in the Event Editor«, section 18 »Fast TRANSFORM«)*.

DELETE DEFAULT the »DELETE DEFAULT« icon imports the cursored event's values into the »MODEL« line from the event list and switches the window's mode to »DELETE«. Setting the Conditions uncovers the desired values prior to deleting.

4.4 »RESULT«

The fourth line down in the TRANSFORM area is the »RESULT« line. This contains the value(s) into which the events' corresponding values are to be transformed if accepted by the »MODEL« and »CONDITION« lines above.

It also determines which values should be further processed by the »VALUE PROCESS« box below.

The result columns have the following meanings:

STATUS if this status is different from the model's, the event's status is transformed. Otherwise, it remains as it was.

Example: transforming Control Change 7 Volume events into P_USER 1 tempo events.

CHA, ONE, TWO if specific values are inserted into any of these three columns, the corresponding values of the accepted events will be transformed into those values.

If the values »Cha«, »ONE« and »TWO« (which are inserted in the columns by default), are left in the corresponding columns, the event's corresponding values will remain untransformed. The same applies if the »*« symbol is inserted in these columns instead.

The advantage of using the »Cha«, »ONE« and »TWO« values is that they can be used to either move one value into the column of a different value, or to send a value into the »VALUE PROCESS« box for further transformation, in which case, the result varies with the original value.

Example: insert »TWO« in the »ONE« column, and insert »ONE« in the »TWO« column. If the status of the accepted event is a Note, then the velocity determines the pitch, and the pitch determines the velocity.

LENGTH for accepted notes, a specific length can be inserted here into which the lengths will be transformed.

If the value »Length« (which is inserted in the column by default), is left in the column, the notes' lengths will remain untransformed. The same applies if the »*« symbol is inserted here instead.

The advantage of using the »Length« value is that it can be used to send a length into the »VALUE PROCESS« box for further transformation, in which case, the result varies with the original length.

TRANSFORM example:

Position	Subpos.	Status	Cha	One	Two	Length
><	*	=	=	=	*	*
1 1 1 1		Control	3	1		
9 1 1 1						
		Control	3	7	TWO	LENGTH

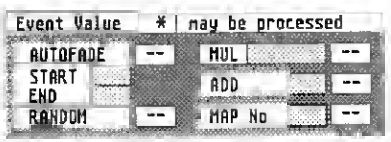
This means: »only accept Control 1 events for transformation if they are on Channel 3 and are situated in the first eight bars of the track, no matter what the *amount* (column TWO)«.

If accepted, the Control 1 Modulation wheel events of this track are transformed into Control 7 Volume events, with the original *amount* of Mod wheel defining the *amount* of Volume.

(The »LENGTH« parameter has no meaning when the status is not »NOTE«).

5. VALUE PROCESS AREA

The result of the transformation can be influenced by the VALUE PROCESS AREA.



The name of the value to be given this variable processing is inserted in the »Event value ??? may be processed« line. The following values are possible:

- CHA, MIDI Channel
- ONE, »first data byte«
- TWO, »second data byte«
- LEN, note length
- POS, event position
- *, the VALUE PROCESS box is switched off.

Important: these settings relate to the »RESULT« line. The meanings of ONE and TWO depend on the result event's status. *(Please read Chapters 8 and 9 for details of the meaning of »first/second data bytes«.)*

The VALUE PROCESS parameters can be freely combined. They are inactive when the box by each name says »---«: click this to switch them on. They have the following meanings:

MUL, multiplication/division data can be multiplied by 999.0000 to –99.0000, with up to four decimal places. Division is the result of multiplying by a value smaller than 1 (eg multiplying by 0.5 divides by two).

Example: multiply all pitch (ONE) values by 2. If the result status is a note, a chromatic scale becomes a whole tone scale.

ADD, addition/subtraction any value can be added or subtracted.

Example: add –12 to all pitch (ONE) values. If the result status is a note, the notes are transposed down an octave.

MAP, the »Universal Maps« click the word »MAP« to enter the »Universal Maps« window, of which there are five user-definable ones. Each of the 128 source values can be allocated to any destination value simultaneously (see section 6).

Example: all C1 notes should become F#3, all B6 notes should become D4 etc, and all at the same time.

AUTOFADE allows linear value changes to be made between the START and END values. The segment length of this transformation should be set by the two »POSITION« locators. If there is no defined segment, the value increments or decrements by 1 from event to event.

Note: AUTOFADE does not necessarily refer to volume or velocity fades. It can fade, up or down, any desired values between the START and END values.

Example: fading the velocity (TWO) of notes down to value 1 over four bars (this could be done more quickly with the »Process Data« function – *see Chapter 6 »Tracks«, section 4.2*).

RANDOM random values can be introduced between the START and END values.

Example: randomizing the pitch (ONE) of a pitched drum during a segment of the song.

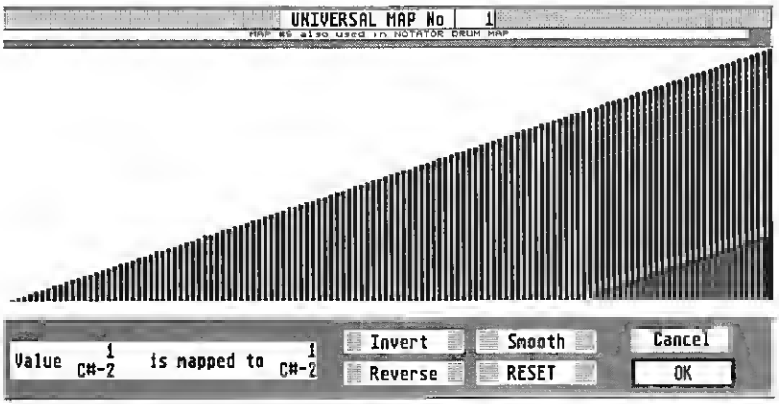
Note: the AUTOFADE and RANDOM parameters are relative to the incoming value – their values are added to the incoming value. To have an absolute result, where the AUTOFADE and RANDOM values replace those of the incoming value, insert a »MUL« value of » .0000« (zero).

6. Universal Maps

6.1 The basics

A »Universal Map« is a table of 128 values (0 to 127). Each source value is given a destination value (»mapping«), and all 128 mappings can take place simultaneously.

The maps default to a balanced configuration: each source value is mapped to an identical destination value so that there is no change. This is seen by the smooth display of beams going up from left to right.



A total of five maps can be programmed, though TRANSFORM can use only one map at a time. Notator users should be aware that Map 5 is used hy the »Drum Map« function in the mapping of note displays on the stave, so keep clear of Map 5 if you are using that particular parameter of »Drum Map«.

6.2 Activating a Universal Map

To activate one of the maps:

In the TRANSFORM window's »VALUE PROCESS AREA«, click the »MAP« parameter to »ON«, select the map number and click the word »MAP«. The Map number is at the top of the map window that appears.

Alternatively, click »Set Universal Maps« (»Options« menu) to gain access from the main page.

The VALUE PROCESS AREA determines which MIDI or P_USER event value is to be mapped *(see previous sections for details)*.

See Chapters 8 »Types Of Event« and 9 »Editing in the Event Editor« for details of the meaning of »first/second data byte« in relation to values ONE and TWO in the TRANSFORM window.

6.3 Entering values

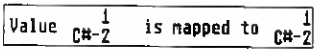
There are many ways to enter values:

- Drag a beam up or down.
- When you first enter a map, click once anywhere in the empty white horizontal bar between the beams and the grey command area. Now you can use the + and − keys in the calculator keypad to alter the destination value, and the left/right cursor keys to select the source value (adding »Shift« speeds up the movement).

  Alternatively, you can scroll the destination value by keeping either mouse button pressed just below a beam.
- Alternatively, destination values can be typed in using the calculator keypad's numbers once the beam is selected by direct beam clicking or use of the cursor arrow keys.

6.4 Display of source and destination values

The »Value (source) is mapped to (destination)« box shows the map's source and destination values. Values are given in MIDI value number terms, and pitch terms in case the value being mapped is a note's »ONE« value.



6.5 Universal Maps: extra functions

6.5.1 RESET

Resets the map to its default configuration of »same source-same destination«, ie neutral.

6.5.2 Smooth

»Smooth« smooths out a curve and evens out jumps and large value differences.

With each left-click, the destination values are quite severely smoothed or ironed out: »hard smoothing«.

With each right-click, the smoothing effect is less, but max/min differences within a region are kept: »soft smoothing«.

»Smooth« is helpful with Velocity mapping where abrupt changes are not wanted (when value TWO is in the »VALUE PROCESS« box if the »RESULT« status is a note).

6.5.3 Invert

»Invert« inverts the curve: 127 becomes 0, 126 becomes 1, 125 becomes 2 ... and so on until 0 becomes 127.

Uses: inverting a keyboard for party tricks (when value ONE is in the »VALUE PROCESS« box if the »RESULT« status is a note). Or having a high velocity when playing softly and vice versa (when value ONE is in the »VALUE PROCESS« box if the »RESULT« status is a note).

6.5.4 Reverse

»Reverse« exchanges source and destination values.

Note: clicking »Reverse« once, then once again does not always re-reverse as you might have thought. If more than one source value was being mapped to just one destination value, the function is unable to know which were the original source functions.

6.6 Saving/loading the Universal Maps

Maps are saved along with song data in a *.SON file. They can be independently imported from any *.SON file using »Load System« *(see Chapter 28 »Disk Operations, section 3).*

7. AUTO NOTEOFF HANDLING

TRANSFORM automatically handles Note Off events when you manipulate Note On events *(see »Appendix«, section 1.7 »One note: two events«).*

The »AUTO NOTEOFF HANDLING« icon switches this Note Off handling off and on. Basically, you should always leave it on.

8. Operating Transform

8.1 TRANSFORMing a track's data

Click the track before opening the TRANSFORM window, then click »TRANSFORM« (»Functions«) menu or press ›Alternate-T‹.

TRANSFORM is also accessible from the event editor.

To carry out the transformation, click »OK«. The window remains on the screen, ready for more commands. If none are required, click »EXIT«.

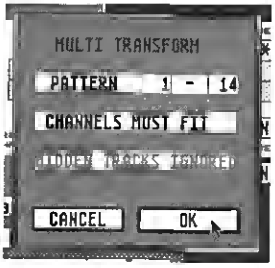
Alternatively, instead of clicking »OK«, press ›Return‹: this carries out the transformation and quits the window.

»EXIT« allows the window to be quitted without any transformation.

8.2 Multi-TRANSFORM (-DELETE, -TransCOPY)

»Multi-TRANSFORM« allows TRANSFORM to be used on more than one track at once. Once a TRANSFORM Set has been prepared, click »MULTI«.

In the Multi-TRANSFORM window, set the following parameters:



- first (»from«) and last (»to«) pattern numbers whose tracks are to be transformed (both numbers are inclusive!).
- if »CHANNELS MUST FIT« is enabled, only those tracks whose »CHANNEL« track parameters match those of the current track will be transformed.
- if »HIDDEN Tracks IGNORED« is enabled, tracks which are hidden with the »HIDE« function will not be transformed *(see Chapter 6 »Tracks«, section 1.5)*.

Click »OK« or press ›Return‹ to carry out the multi-transformation.

This function is able to alter large parts of the song in one go. It makes sense to save the song to disk before using it, in case the result is not what you intended.

8.3 Operating REALTIME TRANSFORM

The large black »REALTIME TRANSFORM« area gives an optical clue that the POSITION, LENGTH, AUTOFADE and RANDOM parameters are not, logically, available in realtime.

Click the »REALTIME TRANSFORM AREA« to enter directly into the »Input Handling« window. Enter the TRANSFORM Set numbers (not Set 0) in each Input box: two sets per Input. If Unitor is connected, up to six Sets could be in simultaneous use.

The action of entering the Set numbers in the »Input Handling« window activates those Sets for realtime use.

Note: it is sometimes important in which order the two Sets are entered in each Input.

8.4 REALTIME TRANSFORM examples

The following examples serve to illustrate the realtime applications of TRANSFORM. You will find it useful to work through them to gain more of an insight into their workings.

Example 1

This example shows how to slightly expand the dynamics of the velocity in realtime, so that weakly-struck notes become quieter and forcefully-struck notes louder (assuming the keyboard is velocity-sensitive).

- The MIDI Thru function must be on, and »Auto-Off Channel« must be off/blank (both in »MIDI Thru« in the »MIDI« menu). The keyboard must be switched to »MIDI Local Off« if it is a synthesizer being used for its own sounds.
- Press ›Alternate-T‹ to enter the TRANSFORM window. Click the black »REALTIME TRANSFORM AREA« to enter »Input Handling«. Assuming the keyboard is connected to Input I, place a »1« in the upper »TRANSFORM No« box. Click »OK«.
- Back in the TRANSFORM window, select Set 1 and click »DEFAULT« to zero it. Check that the status of both »MODEL« and »RESULT« lines says »NOTE«.
- Install TWO in the »Event value ??? may be processed« line (TWO defines the note's »second data byte«: velocity).
- Switch »MUL« to »ON«, install value »1.3000«. Play the keyboard: its velocities have all been slightly increased (once the »Input Handling« window is looking at a Set number, whatever values you install in that Set can be heard live).
- Switch »ADD« to »ON«, install value »–13«. This takes thirteen velocity points away from every note to allow the velocity range to fit within the MIDI 0 to 127 range.

This example helps expand the velocity of, for instance, a DX7 synthesizer's keyboard.

Example 2

You would like to be able to switch between two pre-programmed levels of volume (Control 7) with the sustain footswitch (Control 64).

For REALTIME TRANSFORM you require two Sets installed in the correct MIDI Input box in the »Input Handling« window.

Set 1:

STATUS	CHANNEL	ONE	TWO
=	*	=	*
Control		64	0
Control	Cha	7	96

Set 2:

STATUS	CHANNEL	ONE	TWO
=	*	=	=
Control		64	127
Control	Cha	7	127

Without the footswitch (its TWO value is 0), the volume is 96. When you press the footswitch (TWO value »127« means »on«), the volume is 127.

Example 3

For REALTIME TRANSFORM, install the required Set number (not 0) in the correct MIDI Input box in the »Input Handling« window.

First set the synthesizer's »pitch bend range« to twelve semitones (one octave).

For a »guitar-like« solo, you would like the upwards movement of the pitch wheel to be limited to one tone's bend, but the downwards movement to go down a full octave.

There are two ways of achieving this:

Version 1:

STATUS	CHANNEL	ONE	TWO
=	*	*	><
PitchWh			74
Control			74

Here, the upper end value (two semitones higher) has already been reached 1/6th of the way up.

Version 2:

STATUS	CHANNEL	ONE	TWO
=	*	*	><
PitchWh			65
			128
PitchWh	Cha	ONE	TWO

Value TWO may be processed
MUL: »0.08« and »ON«
ADD: »64«

Here, the upper end value is reached by full movement of the wheel upwards. This is done by compressing the upwards values; all the in-between values are kept.

9. TRANSFORM examples

The following examples are equally suitable both for REALTIME TRANSFORM and normal transformation of existing data in a track. You will find it useful to work through them to gain more of an insight into the workings of TRANSFORM.

Example 1

The Modulation wheel (vibrato) effect is too much for the sound being used in a solo:

STATUS	CHANNEL	ONE	TWO
=	*	=	*
Control	Cha	ONE	TWO

Value TWO may be processed:
MUL: »0.7« and »ON«

The »second data byte« »amount« values are represented by the »TWO«. By dividing by »0.7« values are reduced, eg 50 becomes 35.

Example 2

An existing string of Modulation wheel events need to be transformed into a smooth Volume fade-in during the first four bars.

Remember that events have to exist before they can be transformed: TRANSFORM does not produce events »out of thin air«, unlike the Realtime MIDI Generator or Hyper Edit.

POSITION	SUBPOS.	STATUS	CHA.	ONE	TWO
><	*	=	*	=	*
1 1 1 1		Control		1	
5 1 1 1		Control	Cha	7	TWO

Value TWO may be processed:
AUTOFADE: »ON« MUL: 0.0000 ON
START: 0
END: 128

By multipling by »0«, you ensure that, whatever the original values of the Mod wheel events, their »second data bytes« are *replaced* by a linear change between 0 and 127.

The smoothness of the fade depends on how many events there are between the two POSITION locators. The more events there are, the smoother the effect.

Example 3

Changing a Mod wheel movement into Tempo (P_USER 1 events).

STATUS	CHANNEL	ONE	TWO
=	*	=	*
Control		1	
P_USER		1	TWO

Value TWO may be processed:
MUL: »2.0« und »ON«

Control 1 events are transformed into P_USER 1 (»absolute tempo«) events. Their »second data bytes« (TWO) are doubled to take advantage of the full range of tempo, which can go up to 250 bpm, whereas normal »second data bytes« only go up to 127.

Example 4

Changing all MIDI Channel 3 notes to Channel 5.

STATUS	CHANNEL	ONE	TWO
=	=	*	*
NOTE	3		
NOTE	5	ONE	TWO

Example 5: »DELETE«

Deleting all notes whose Channels are 10 to 14 inclusive. This can also be done in realtime.

You can either enter the TRANSFORM window and click the »DELETE DEFAULT« icon to put the window in »DELETE« mode, or enter the window via »Delete/Keep Events« (»Functions« menu), in which case the window will already be in »DELETE« mode:

STATUS	CHANNEL	ONE	TWO
=	><	*	*
NOTE	10		
	15		

(there is no »RESULT« line in »DELETE« mode)

Example 6: »keeping«

Keeping all C3 notes on Channel 1 if they are above velocity 90 (in other words, delete all other notes). »DELETE« mode:

STATUS	CHANNEL	ONE	TWO
=	<>	<>	<>
NOTE	1	C3	90
	2	C#3	127

(there is no »RESULT« line in »DELETE« mode)

Here, you are deleting all events whose Channels are »outside« 1, whose pitches are »outside« C3 and whose velocities are »outside« the 90 to 127 zone.

Example 7: mapping a drum track

You have a drum track containing a number of drum notes that relate to your own drum machine. A friend brings in a drum machine set to a different note/drum configuration which you want to use for this song. Neither of you wants to change the internal parameters of either machine. Use TRANSFORM with a Universal Map:

STATUS	CHANNEL	ONE	TWO
=	*	*	*
NOTE			
NOTE	Cha	ONE	TWO

Value ONE may be processed:
MAP 1: »ON«

Your drum machine's notes are the source. Your friend's machine's notes are the destination. Map each drum to the same drum destination in your friend's machine. When all the drums are mapped, click »OK« in the Map, then press ›Return‹ in the TRANSFORM window.

Example:

Snare – map C1 to C3
Kick – map D1 to G3
Tom1 – map B1 to D4 etc

10. Saving/loading the TRANSFORM Sets

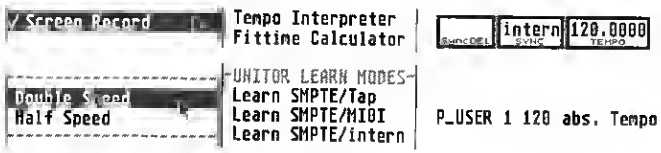
The eleven Sets including all their parameters form part of a song's data and are saved in a *.SON file. »Load System« *(see Chapter 28 »Data Management«, section 3)* allows them to be imported separately into an exisiting song from any *.SON file.

11. Transformations overview

Desired transformation		*Use this parameter*
Reduction	a) by division	»MUL« (less than 1)
	b) by a set amount	»ADD« (minus)
Increase	a) by multiplication	»MUL« (greater than 1)
	b) by a set amount	»ADD« (positive)
Compression/Expansion	a) to a set amount	»RESULT« line
	b) linear	»MUL«
	c) optional	»MAP« (curve)
Change over time	a) swell up/down	»AUTOFADE«
	b) random	»RANDOM«
Conversion	c) user-defined	»MAP«
Inversion		»MAP« (»Invert«)
Allocation	a) optional	»MAP«
	b) direct	»RESULT« line
	c) random	»RANDOM«
	d) swap columns	eg »TWO« to »ONE«

Chapter 25
Tempo

25



1. The basics

To ensure that recorded notes and other events are arranged in precisely the right order, any sequencer must have a form of internal metronome or »time keeper«: this is the sequencer's »clock«, which, like all clocks, gives out ticks or »pulses«, though these pulses are internal and not heard by the user.

The closer its pulses are to each other (called the »resolution«), ie the more pulses there are in a given unit of time, the more accurately the sequencer can time the position of an incoming event. An incoming event is assigned to the nearest clock pulse.

Normal analogue tape recording represents the highest resolution you can ever get, because an analogue tape recorder does not have to use a clock to time notes.

To use the built-in advantages of the sequencer, such as »quantization« (time correction), the pulses relate to whole fractions of musical units. As a consequence, the user must play in time with the metronome.

Only then can the sequencer recognize the user's intended timing, and only then can the positioning or altering of events' time positions be expressed in terms of bars, beats etc.

The speed of the clock is governed by Creator/Notator's »tempo«.

See also »Appendix«, section 1.8 »Resolution: display in notes or pulses per quarter note?«, and Chapter 5 »Positioning«, section 2 »Internal clock resolution ...«.

2. Tempo without changes

The tempo display is in the »Tempo« box in the »information bar« over the top of both the main page and event editor. In »Internal Sync« mode, it can be set from 25 to 250 beats-per-minute (bpm).

It can be clicked with either mouse button, or altered using »mouse as slider« or »Control-&-mouse« *(see Chapter 3 »Concepts«, section 1)*.

The tempo is valid for the entire piece of music in the sequencer if there are no »tempo changes« made by »P_USER 1« events (see below).

The tempo display can be set with the precision of up to four decimal places, for example 120.0001 bpm. From a musical point of view this precision is quite unnecessary, since the difference between, say, 120.0001 and 120.0002 is not perceptible.

Note: in »Internal Sync« mode, the tempo resolution goes as far as two decimal places, eg 120.01.

The point in having such a high tempo resolution is to allow synchronization of the music to external synchronization sources to take place to the nearest millisecond (or better), which is possible with the Unitor. Examples: video post-production, or the simulation of other synchronizers' timing characteristics *(see also Chapter 27 »Hardware Peripherals: ...«, section 9 »Editing a »Sync Reference«)*.

Internal tempo changes cannot take place in »MIDI Sync« mode: these are governed by the external synchronizer, though external tempo changes will be displayed in the »Tempo« box *(see Chapter 26 »Synchronization«, section 4 »MIDI Sync« mode«)*.

In »MIDI Sync« mode, it does not matter if Creator/Notator's tempo display shows a slightly different tempo to that of the external synchronizer, or even moves about *(see »Appendix«, section 1.6 »Clock »Interpolation«...«.)*.

3. Tempo changes via »screen recording«

When »Screen Recording« (»Flags« menu) is ticked, tempo changes are possible by altering the tempo display during Record mode.

Click an empty track, click »RECORD« and alter the tempo display using the mouse. You can also use the »mouse as slider« function, or the »Control-&-mouse feature *(see Chapter 3 »Concepts«, section 1 »Mouse«)*.

›Control‹ & mouse has an additional feature for use with screen recording: if the ›Shift‹ key is held at the same time as ›Control‹, changes to the tempo are not recorded until ›Shift‹ is released, allowing realtime jumps in tempo to be made.

Tempo changes are recorded as »P_USER 1« (absolute tempo) events in the current track *(see also next Chapter)*.

P_USER 1 (absolute tempo) events are »TRUE« events *(see Chapter 5 »Positioning«, section 3.2 »TRUE Tempo«)*.

4. Tempo changes via event editor

P_USER 1 events allow you to pre-program a completely new tempo at any time position by scrolling their »second data bytes«.

To do this, drag a P_USER event from the »PSEUDO« icon into the event list *(see Chapter 9 »Editing in the Event Editor«)*. Ensure the P_USER event says »1« in the »–1–« column. The »–2–« column contains the tempo value, in whole bpms.

Tip: tempo changes with decimal places must be created using the »Fittime Calculator« (see section 5 below).

Example:

1	**1**	**1**	**1**	P-USER	1	120	abs. Tempo
4	1	1	1	P-USER	1	130	abs. Tempo

Here, the tempo is 120 bpm at the beginning of the pattern, changing to 130 bpm at the beginning of the fourth bar.

P_USER 2 events are »relative tempo« commands that add to or subtract from the main tempo.

P_USER 2 events will only work in »Internal Sync« mode.

They can only be entered manually into the event list (no screen recording), and their use is limited to »Internal Sync« mode. The »second data byte« in column »–2–« of the event list gives the number of bpms added to or subtracted from the main tempo.

P_USER 2 events are useful in situations where the ratio of tempo changes is to remain constant, even if the main tempo is altered. P_USER 2 events are not »TRUE« events.

Tip: keep tempo changes in their own track. This makes it easier to see what your tempo programming is, and helps when using the »Create Using Tempo Of …« function in the »Sync Reference« window *(see Chapter 27: »Hardware Peripherals: …«, section 8.2)*.

5. »Fittime Calculator«

Tempo changes using up to four decimal places can be inserted with the »Fittime Calculator«. This is of special use during »SMPTE Sync« mode synchronization in video etc post-production *(see Chapter 27 »Hardware Peripherals: ...« , section 13 »Fittime Calculator«)*.

You may use the Fittime Calculator to insert P_USER 1 events which have decimal places. This is not possible by the normal process of dragging P_USER 1 events into the event list *(see section 4 above)*. To do this, ensure the upper segment display is set to the desired time position, select the tempo and click »INSERT Tempo CHANGE«. In »Internal Sync« mode, up to two decimal places are recognized by the program.

6. Tempo changes via the »Tempo Interpreter«

Tempo changes can be created and recorded by the analysis of external timing from:

- incoming MIDI notes *(see Chapter 26 »Synchronization«, section 6)*, and
- incoming acoustic sounds with Human Touch *(see Chapter 26 »Synchronization«, section 6 and Chapter 27 »Hardware Peripherals: ...« , section 14)*.

6.1 Tempo changes while in »SMPTE Sync« mode

One special use of the »Tempo Interpreter« is the automatic simulation (analysis, recording, optimization) of the timing characteristics of other synchronization devices *(see Chapter 27 »Hardware Peripherals: ...« , sections 10.1 »Learn SMPTE/Tap« and 10.2 »Learn SMPTE/MIDI«)*.

7. Quasi »tempo changing« by altering notes' time positions

The following functions do not affect the tempo display. Instead, they alter the relationships of note positions.

7.1 »Double Speed«

The »Double Speed« function (»Functions« menu) halves the distance between notes, resulting in the track playing at twice the speed.

»Double Speed«, used on 1/8ths at tempo 120 does not play the 1/8ths at 240 bpm, but alters the notes to 1/16ths (still at tempo 120).

Note: the internal clock resolution cannot be doubled to match: if you do »Double Speed« then »Half Speed« to a track, the distances between notes will be rounded off to the nearest 1/384th notes.

Tip: try playing in an arpeggio in 1/8ths, then use »Double Speed« once, then once again. This results in a 1/32nds arpeggio.

7.2 »Half Speed«

The »Half Speed« function (»Functions« menu) doubles the distance between notes, resulting in the track playing at half the speed.

Because the track doubles in length, tracks which are already very long may go over the maximum allowed number of bars (approx 1350).

7.3 Using »Transform« to »change tempo«

The »Double Speed«/»Half Speed« functions double/halve the number of pulses in the note position values. The »Transform« function *(see Chapter 24 »Transform«)* can use other multiplication and division factors to create various «tempi« for each track. Because the calculations can use four decimal places, subtle differences can be achieved.

A multiplication factor greater than 1 spreads a track out, thereby slowing it down. A factor less than 1 compresses it, making it quicker. Because »Transform« can limit changes to segments of a track, tempo changes within a track can be made without touching the tempo display.

Example: install the following in the «Transform« window:

As with the »Half Speed« and »Double Speed« functions, the zero point of the calculations is position »1 1 1 1«.It follows that with segments that do not start on this position, apart from compressing or stretching the events' relative time positions, these calculations will also move the segment's start position . If this is not desired, »Segment Copy« the segment to «1 1 1 1« (if necessary in another track), »Transform« it, then replace it where it belongs. To move segments within a track, use »Insert Mode« *(see Chapter 9 »Editing in the Event Editor«, section 5.1.2 »Insert Mode«).*

Chapter 26

Synchronization

26

1. The basics

Synchronization is the »locking together« of two devices that each have their own internal clocks, so that they run »in sync« together as if they were one device. This is rather like a car (device 1) towing a trailer (device 2). You normally talk about the »master« (the car) and the »slave« (the trailer).

Examples of common synchronization:

sequencer master – drum machine slave
sequencer master – sequencer slave
synchronizer master – sequencer slave
video recorder master – audio tape recorder slave etc.

With Creator/Notator, if it is the slave of another device (synchronizer, other sequencer etc) we talk about »external synchronization«, and the »Sync mode« *(see below)* will be set to »MIDI Sync« mode.

With external synchronization, the Start point and tempo are determined by the other device.

This is so Creator/Notator react quickly to commands.

Synchronization is conducted by regular »MIDI Clock« pulses, sent by the master through the MIDI cable whenever it is running.

These MIDI clock pulses are read by the slave, but invisible and inaudible to the user. Each MIDI clock pulse advances the slave by 1/96th note.

2. Choosing the »Sync« mode

Y Choose the »Sync« mode in the »SYNC« box in the »information bar«. Pressing ›Y‹ or clicking the box gives you:

»internal« the sequencer is not slaved to any external device and runs according to its own tempo.

»MIDI« the sequencer is slaved to an external device, which determines Start point and tempo.

»*SMPTE*« the sequencer runs according to its »Sync Reference« *(Chapter 27 »Hardware Peripherals …«, section 6)* using Unitor as the synchronizer. Here, Creator/Notator and Unitor are a »partnership«, not »master« and »slave«.

»*manual*« the sequencer is slaved to incoming MIDI notes and/or acoustic signals, which determine the tempo and Start point according to the »Tempo Interpreter« function *(see Chapter 27 »Hardware Peripherals: …« sections 10 and 14).*

Recording can take place whatever the »Sync« mode.

3. »Internal Sync« mode

In the »Internal Sync« mode, Creator/Notator's own internal clock determines the tempo: it is »its own master«. Incoming MIDI Clock pulses are ignored *(see also Chapter 25 »Tempo«)* since »Clock In« (»MIDI« menu) is automatically disabled.

4. »MIDI Sync« mode

In »MIDI Sync« mode, Creator and Notator react to all »MIDI System Realtime« and »System Common« messages: »Clock In« (»MIDI« menu) is automatically enabled.

These are »Start«, »Stop« and »Continue«; the MIDI Clock pulses that advance the sequencer by 1/96th note per pulse; and the regular »Song Position Pointers« that give information on where the master is in the music, relative to the »1 1 1 1«. Creator and Notator are able to react to a new position via »Song Position Pointers« whether in Record or Playback modes.

If »Position Pointers« and »Clock In« (»MIDI« menu) are disabled, incoming MIDI Sync data will be ignored.

4.1 »Interpolation« during »MIDI Sync« mode

Although the standard MIDI Clock's resolution is just 1/96th note, Creator/Notator retain their 1/768ths (1/1536ths) resolution, even in »MIDI Sync« mode. The resolution of the sequencer is therefore eight (sixteen) times higher than that of the MIDI Clock.

This is done by something called »Clock Interpolation« *(see »Appendix«, section 1.6 »Clock »Interpolation«…«)*. Creator/Notator display the tempo even in »MIDI Sync« mode; if this tempo display is different from that of the master's, it has no bearing on the precision of the synchronization, which is totally locked to the master.

4.2 Simple example of a »MIDI Sync« mode situation

– Connect the MIDI Output of a drum machine to the Atari MIDI Input.
– Switch Creator/Notator to »MIDI Sync« mode.
– Start the drum machine.

The drum machine controls the sequencer's Start and Stop points and its tempo.

4.3 MIDI synchronization devices

To synchronize the sequencer to a tape recorder, a synchronizer is needed. It allows the MIDI world to »lock to« the acoustic world of wind instruments, voices etc which are usually recorded on tape. The synchronizer ensures that the sequencer runs at the same speed to and parallel with the tape.

You have basically two choices of synchronizer:

– the usual external synchronizer that reads timecode and translates it into »MIDI Clock« pulses and »Song Position Pointers«: it is used with Creator/Notator in »MIDI Sync« mode.
– Unitor.

Using Unitor to synchronize to an audio or video tape recorder etc means a particularly accurate and easy way of working.

Unitor is not the usual type of »SMPTE-to-MIDI synchronizer«: it bypasses MIDI Clock altogether by communicating directly with the computer, making it much faster than a »normal« synchronizer that has to translate SMPTE into MIDI Clock pulses – its response rate is some 1000 times quicker *(see Chapter 27 »Hardware Peripherals: ...«)*.

External MIDI synchronizers divide into two camps:

– those that read SMPTE timecode, and
– those that do not (but read FSK or other timecodes).

These days, synchronizers that do not use SMPTE usually use a method similar to SMPTE to allow starting halfway through the music; they are also slightly cheaper than SMPTE synchronizers. However, their codes are not universal and cannot be read by other makes of synchronizer.

SMPTE timecode, recorded on one of the tape's tracks, tells a synchronizer what the tape position is at the tape machine's replay head, in terms of absolute units of time *(see Chapter 27 for more SMPTE details)*.

The synchronizer translates the SMPTE position into musical terms (bar, beat etc) and sends the data as »Song Position Pointers« to the sequencer. The sequencer positions itself accordingly and runs in sync with the tape, locked by the MIDI Clock pulses. SMPTE synchronization means being able to start the tape at any position and the sequencer will react after about two seconds (Unitor is much faster!).

The user must program certain information into the external synchronizer: the Start time at which the sequencer should Start (sometimes called the »SMPTE Offset«); the Start tempo, and any tempo changes; the SMPTE frame rate.

Read your synchronizer's manual carefully.

On the whole, SMPTE synchronization is reliable, though video tape recorders other than »U-Matics« are prone to dropouts *(VITC code is more reliable: see next Chapter on Steady Eye).*

To be able to record in the sequencer when using an external synchronizer, the synchronizer must have a MIDI In port to attach to the master keyboard's MIDI Output. This is because the Atari's MIDI Input is occupied by the cable from the synchronizer's MIDI Output.

Note: external synchronization via MIDI Clock pulses does not allow positions earlier than bar 1. The »Song Position Pointers« do not allow negative values, and the MIDI »Start« command includes a command to »jump to bar 1«.

The actual *internal* start point within the sequencer can always be freely decided. For instance, the first entry of the arrange list could have a start-bar of »2«, which would allow a one-bar count-in at the beginning of the piece, even in »MIDI Sync« mode.

4.4. Recording in »MIDI Sync« mode

Creator/Notator can record during »MIDI Sync« mode.

»STOP ends Record« in »MIDI Thru« (»MIDI Menu«) determines whether stopping the tape should automatically end a recording *(for details on »Stop ends Record« please read Chapter 21 »Overall Program Settings«, section 4 »STOP ends Record«).*

4.5 Recording data from other MIDI sequencers

Music can be recorded from any other MIDI sequencer. Do as follows:

1. Connect the other sequencer's MIDI Out port to Creator/Notator's MIDI Input.
2. Switch Creator/Notator to »MIDI Sync« mode.
3. Select an empty track and click »RECORD«.
4. Start the other sequencer.

So long as the other sequencer is issuing MIDI Clock pulses and starts with the correct MIDI Start command, all the notes and events will be recorded into Creator/Notator's selected recording track. The MIDI Start command includes the command »jump to bar one«, so Creator/Notator will ignore any count-in and go straight to its »1 1 1 1« on receiving the command.

If you need to record segments which are not at the start of the piece (ie any segment that does not start at »1 1 1 1«), use the Main Bar Counter to select the segment's start point in Creator/Notator and click »DROP«; in the other sequencer, select the same segment start point, and start it with its »CONTINUE« command.

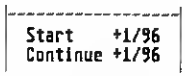
The tempo at which the transfer takes place is not so important, and can be altered subsequently. The quicker the tempo, the sooner you will be finished; the slower the tempo, the more accurate the recording will be where large quantities of data are involved or where timing is critical. Where the transfered data was quantized in the other sequencer, it should be quantized again in Creator/Notator using a small amount of time correction, say, »QUANTIZE« »96« to ensure that all is well with the timing.

In general, there tends to be a delay of one to two pulses in the transmitted data (you can see this in the event list). Instead of quantizing, you may prefer to use »Insert Mode« to move the whole track earlier *(see Chapter 9 »Editing in the Event Editor«, section 5.1.2)*.

After the transfer, you will be left with one very long track and a lot of free tracks in Creator/Notator, though this will sound no different than before. If you want to do further work on the music, use »Demix All Channels« *(see Chapter 18 »Copy, Merge ...«, section 7.1)* to split up the data in the recorded track into its constituent MIDI Channels, but this can only be done if the other sequencer's tracks were assigned to different MIDI Channels before the transfer. You should do this, even if you did not actually use all those Channels to control instruments from the other sequencer. Alternatively, you could transfer the data track by track if you have the time and the patience to do so.

4.6 »Start/Continue +1/96th note«

»Start +1/96 Note«, »Continue +1/96 Note« (»MIDI« menu): these should not be ticked unless there is reason for doing so. Unticked, they conform to MIDI protocol.



If enabled, Creator/Notator add one MIDI Clock pulse (1/96th note) to the incoming MIDI Clock data at each Start or Continue command they receive when in »MIDI Sync« mode (so they will always be 1/96th note ahead of the master device).

These options were conceived for devices which are now consigned to the history books.

5. »SMPTE Sync« mode

»SMPTE Sync« mode is reserved for Unitor *(see Chapter 27 »Hardware Peripherals: ...«)*.

6. »Manual Sync« mode and the »Tempo Interpreter«

In »Manual Sync« mode, the sequencer's Start point and tempo are calculated as a result of the realtime analysis of time positions of incoming MIDI notes. If Human Touch is connected, incoming audio or click signals can be analyzed in the same way.

The synchronization here is relatively »loose«, ie if the external sync signals stop, the sequencer can continue under its own steam using its internal Clock. The »Tempo Interpreter« controls what happens live in »Manual Sync« mode.

The »Tempo Interpreter« can also be used in conjunction with Unitor via the two »Learn Modes« »MIDI/SMPTE Learn« and »Tap/SMPTE Learn« (»Options« menu). Here, the tempi of the incoming MIDI notes or audio signals are set against the incoming SMPTE timecode to create a »Sync Reference« *(see next Chapter)*.

6.1 The basics

The »Tempo Interpreter« stands the normal situation on its head: it's not you having to play to the sequencer, but the sequencer that follows your tempo. The use of Human Touch even allows the tempo to follow either acoustic instruments via microphone or audio signals off tape.

In »Manual Sync« mode, Creator/Notator will automatically set the tempo according to a realtime analysis of the timing of incoming MIDI notes or audio signals received by Human Touch. The »Tempo Interpreter« can read tempo and sync information from even relatively complex signals and rhythms. You do not have to play exact quarter/eighth etc notes – even chords and a rhythmical style of playing will be acceptable to the »Tempo Interpreter«, and any pauses will be bridged.

The basis of how it works is as follows: you enter a »Tap Step« of, say, a quarter note into the »Tempo Interpreter« window. Like any musician, it can then work out approximately when it can expect the following quarter notes. Every »tap« or signal that falls outside that expectation will be ignored. If the quarter note tap/signal arrives more-or-less on time, the »Tempo Interpreter« will accept it and alter the tempo to suit. If the

tap came a little early, the tempo will increase;if late, the tempo will decrease. How »forgiving« the »Interpreter« is in accepting tap inputs is defined by the »Window« setting. For example, 8/768 (= 1/96th note) means that the timing of the tap can waver up to 1/96th note either side of the »ideal« quarter note.

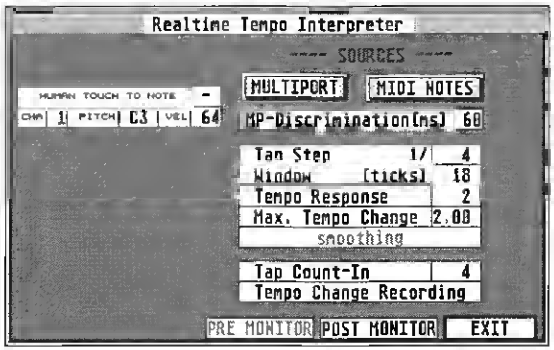
Especially where the taps are relatively imprecisely played in by hand, it makes more musical sense to »buffer« the response of the Creator/Notator tempo by selecting smaller values in the »Tempo Response« box; you may even limit the amount by which the tempo can change (»Max Tempo Change«).

6.2 »Tempo Interpreter« window

The »Tempo Interpreter« is activated by switching over to »Manual Sync« mode (press ›Y‹).

The »Tempo Interpreter« can be addressed by playing any MIDI keyboard or via Human Touch.

To open the »Tempo Interpreter«: click in the »Options« menu, press ›Alternate-I‹ or left-click the »OVERFLOW« box below the pattern window.



Like »CONTINUE«, a Start command via the »Tempo Interpreter« continues from the current position of the sequencer. In most cases, you will need to ensure that the Main Bar Counter is set to »1 1 1 1« (eg by pressing ›Enter‹ twice).

If there is a value in the »Tap Count-In« box, the sequencer will wait for the given number of taps before starting at the interpreted tempo.

SOURCES the two icons »MULTIPORT« and »MIDI NOTES« select the source of the tempo information. They can be active at the same time to allow combined sources.

If »MIDI NOTES« is active, the timing of every incoming note is analyzed.

If »MULTIPORT« is active, the timing of audio or click signals via Human Touch is analyzed *(see Chapter 27 »Hardware Peripherals: ...«, section 14).*

MP-DISCRIMINATION applies only to »MULTIPORT« signals. It shows the number of milliseconds that have to pass after an audio signal, before the »Tempo Interpreter« will accept a new one ie signals that follow on too quickly will be ignored *(see also Chapter 27, section 14).*

The following parameters allow the conditioning of the tempo response. The more rhythmically complicated the source of taps or signals, the more care has to be taken in setting these parameters:

TAP STEP sets the nominal step timing of the inputted tap information. It has been shown that a larger, rather than a smaller »Tap Step« value gives better results: 1/4th note steps is a setting that suits most situations. When altering the value, the »Window« value automatically defaults to a suggested setting.

WINDOW the width of the window is in pulses (1/768th [1/1536th] notes). Tap inputs will be accepted for tempo interpretation only if the incoming taps fall within the »Window« range (the value shows the range either side of the ideal, exact tap step). Taps that fall outside the range (ie which probably make no musical sense) will be ignored. The narrower the window, the more the »Tempo Interpreter« will resist the tempo influences of taps that come in between the »official« steps. If the Window is set wider, then stronger, even abrupt tempo changes can be accepted.

Normally, there is a »fly-wheel« effect: the sequencer will continue playing at the last-received tempo value if the taps stop coming in. If you click the name »Window« to disable the function: the sequencer will immediately stop at the nearest 1/16th note if the taps stop coming in.

Tempo RESPONSE is where you set the sensitivity to tempo changes. It is helpful to reduce this sensitivity to around »2« if the timing of the taps is relatively imprecise.

MAX Tempo CHANGE sets the maximum amount by which the tempo is allowed to change *at each tap.* In very critical situations (eg where a mixed piece of music is being taken as the master tempo-giver via Unitor/Human Touch), it can make sense to take time to ascertain as accurately as possible the tempo of the piece, to set this tempo in Creator/Notator's »Tempo« box in the main page, and to use »Max Tempo Change« to limit any changes to the barest minimum (providing, of course, that the music being used as master tempo-giver has no tempo changes of its own).

SMOOTHING should be used in a »live« situation, since jumps in tempo are smoothed out. When synchronizing to tape using »Manual Sync« mode, »Smoothing« should be switched off by clicking the name to grey, so that the sequencer is as close-coupled to the master as possible.

Tempo CHANGE Recording a switch which allows the tempo changes to be recorded (as P_USER 1 events) in the current track if in Record mode.

PRE MONITOR/POST MONITOR it can help to have an acoustic »click« over the computer monitor's loudspeaker, especially incritical applications involving Human Touch inputs.

If »PRE MONITOR« is enabled, every signal input, whether it falls inside or outside the »Tempo Interpreter« window, generates a high click.

If »POST MONITOR« is enabled, only those signals that fall within the »Window« will receive a low click.

An acoustic input via Human Touch is at its optimal level when you are able to hear, via the PRE MONITOR clicks, a clear indication of what the tempo is doing. Always try to produce the clearest possible tempo signals to avoid spurious information being received at the »Tempo Interpreter«.

There is also a visual feedback on the main page:

- PRE MONITOR signals in the »MIDI THRU« box.
- POST MONITOR signals in the »SMPTE« box. Both boxes are below Creator/Notator's pattern window.

TAP COUNT-IN like a live musician, Creator/Notator can be instructed to wait for a tapped count-in before playing. This count-in also gives the tempo setting for the start of the piece. The normal tempo setting on Creator/Notator's main page will be overruled by the tempo you have given during the tap count-in.

Example: Tap Count-In 4, Tap Step 1/4 means that four quarter notes must be played in, and the sequencer will start on the fifth. First reset the Main Bar Counter by pressing »Enter« twice.

If the Tap Count-In value is set to »zero« (blank), the very first tap/audio signal will start the sequencer (equivalent to the »Continue« command). The start tempo, in this case, is the one in Creator's main tempo display.

Human Touch TO NOTE an incoming audio signal or click can be made to trigger any MIDI note of any connected synthesizer *(see Chapter 27 »Hardware Peripherals: ...«, section 14.3).*

6.3 Examples of the »Tempo Interpreter«

Example 1: the sequencer must be counted-in via a MIDI keyboard, so as to give the correct tempo. Set the following parameters in the »Tempo Interpreter«:

Tap Step – 4
Tap Count-In – 4
MIDI NOTES – On
Window – 0

Play five single notes at the desired tempo on the MIDI keyboard. If you do not wish to start with this method each time, now press ›Y‹ to switch over to »Internal Sync« mode.

Example 2: the tempo is to be influenced by your playing while the sequencer is playing. You want to play with chords note values that are not exact quarter notes; add the following values to the above example:

Window – 18
Tempo Response – 2
Max Tempo Change – 1

Example 3: you want the tempo to exactly follow your tapping as you slow down and speed up. Play single notes only, and play carefully and clearly:

Window – 100 approx
Tempo Response – 8
Max Tempo Change – 9.99

7. MIDI Clock data at the Output

In every »Sync« mode, Creator/Notator send the standard »MIDI System Realtime/Common« commands to one of the Outputs: Start, Stop, Continue, MIDI Clock pulses and »Song Position Pointers«.

The »CLOCK PORT« chosen to transmit this data is selected in the »MIDI Thru« window (»MIDI« menu, or click the »MIDI THRU« box below the pattern window).

Other drum machines, MIDI sequencers etc can therefore be slaved to Creator/Notator. Not all devices can react to changes in position while running, since MIDI Protocol usually only allows positioning in stop mode.

7.1 Filtering the MIDI Clock at the Output

All the »MIDI System Realtime« messages can be prevented from being output by deselecting »Clock Out« (»MIDI« menu). Normally, leave these enabled.

There is no point in disabling the »MIDI Clock Out« thinking this improves »MIDI timing«. You will save only one byte per 1/96th note (1/3000th second or 320 microseconds), which is one third of the time taken by a Note On event: at a moderate tempo, this means one hundredth of a 1/96th note or 1% of the capacity of MIDI. Clocks are in any case strictly regular and cannot lead to irregular variations in timing.

Disabling »MIDI Clock Out« stops a drum machine jumping into »Play« if all you want to do is use it as a source of drum sounds.

If a second sequencer is slaved to Creator/Notator using cables going in both directions, disabling Creator/Notator's MIDI Thru function (»MIDI« menu) prevents MIDI »feedback« within the loop; the second sequencer would otherwise try to send its own commands on receiving Creator/Notator's.

7.2 Delaying the MIDI Clock at the Output

»Sync DELAY« in the »information bar« along the top of the screen determines by how much MIDI Clock pulses at Creator/Notator's MIDI Output are delayed (positive values) or pushed (negative values).

The value is shown in 1/768th notes, though this changes to milliseconds with »Delay in ms« (»Flags« menu).

This moves slaved devices. Some older devices (eg drum machines) need »pushing« because they react slowly to MIDI Clock pulses: this problem is solved by entering a negative value in the »Sync DELAY« box so the slaved device is brought into time. However, MIDI Protocol does not allow a Start command be be transmitted prior to bar 1: so what actually happens is that there is a quick acceleration of the MIDI Out Clocks pulses at the Start, in line with the selected negative value.

8. Fostex MTC-1 support

Normally, when synchronizing a MIDI sequencer to a tape machine, the sequencer is slaved to the timecode coming from one track of the tape via the appropriate synchronizer (eg Unitor).

The Fostex R8 tape machine, together with other Fostex tape machines of the same series (from now on we refer to the »R8« only), alters this scenario, in that it is capable of being remotely controlled via MIDI, using the Fostex MTC-1 synchronization unit:

a software program containing the appropriate commands is able to control the tape machine's transport and record functions, as well as other functions. One of the major advantages is the ability to locate the tape position according to bar/beat as well as SMPTE time.

Notator/Creator Version 3.1 support the Fostex MTC-1, allowing control of the R8 from the main page and RMG.

NB: this operation does not use »MIDI Timecode« (»the MIDI version of SMPTE«).

8.1 Hardware and connections

For successful remote control, you must have both the Fostex MTC-1 and the Unitor, as well as the ST computer (with one Megabyte RAM minimum), Creator or Notator, and the Fostex R8 or similar. One tape track must be striped with SMPTE LTC time-code.

Connections: connect the MTC-1 to the tape machine. Using your mixing desk's patchbay, or a Y-connector, connect the code coming from the timecode track both to the MTC-1 LTC IN port and Unitor's SMPTE IN port. Connect one of the MIDI ports A to F to the MTC-1 MIDI IN port, and switch the »Clock Port« in Notator/Creator's MIDI Thru window to the port you are using.

8.1.1 Signal flow

Many of Notator/Creator's commands additionally transmit MIDI System Exclusive data to the MTC-1. These messages concern the transport commands, as well as information regarding the sequencer's position. The MTC-1 interprets these messages and sends them to the R8 which immediately obeys: eg if the Play command is received, the timecode on the tape track is sent to Unitor and to the MTC-1, a process which guarantees the best possible synchronization between sequencer and tape machine.

8.1.2 Preparations

The Song in Notator/Creator must, as normal, contain a Sync Reference, and the sequencer must be set to »SMPTE LTC« sync mode. Keystroke ›Shift-F‹ activates a special »Fostex mode«. The information is saved with a Song.

Place the tape machine or its detachable command panel within view (though not necessarily within reach!).

8.2 Tape transport

In »Fostex mode«, Notator/Creator's transport commands enter a form of »Local Off« status: pressing »Start« on the sequencer starts the sequencer by issuing a Play command to the R8, setting the tape in motion, which has the effect of setting the sequencer in motion because it is in SMPTE Sync mode and is receiving timecode via Unitor; in other words, the »Start« command does not start the sequencer internally: only via the Fostex.

In Fostex mode:

»Start« starts the Fostex from current TAPE position (like »Continue«).

»Stop« stops the tape. Using »Stop« again rewinds the tape to the beginning.

Positioning: almost every positioning function will control the tape: clicking in the GAM's bar ruler; left-clicking a locator; using »<<<« and »>>>« to make the tape location jump to the next arrange entry; »<<« and »>>« will make the machine fast-forward/reverse while the mouse button is being pressed – if the sequencer and tape were already playing, you will hear the music »cue« forwards or backwards; clicking the Main Bar Counter; recalling stored locator positions via the Function keys, etc.

While the tape is spooling, the sequencer is in Stop mode. The tape will go to the desired position minus the pre-roll (count-in) time, to ensure that the beginning of the desired section is played.

Some operations automatically drop Notator/Creator out of SMPTE Sync mode back into Internal Sync for working without tape: the »GAM Scrub mode«, the Continue command and the event editor CUE function.

8.3 Record

To put a Fostex track into Record Ready status, click the corresponding Creator/Notator track number while holding ›Shift‹. A right-click puts the track back into Safe status. Pressing ›Alternate‹ while clicking a track puts that track into Record Ready status and simultaneously puts all other tracks into Safe status.

Then, to route the signal to the Input Monitor of the selected tracks, press ›*‹ or click »Record« while pressing ›Shift‹. Using »Record« again starts the recording from the tape's current position. If already in Play mode, the Record command allows manual punching in/out to take place.

Particulary useful is the combination of Cycle and Autodrop (enable both modes!), as punch in and out points can be preprogrammed. The desired amount of »pre-roll« can

be set in the Count-in window (»Option« menu). After a successful »Cycle Autodrop« operation, the R8 will drop out of Record mode an grounds of safety.

Tip: depending on the Fostex model, the drop-out command may be slightly delayed. To avoid recording over material that follows the drop-out point, make the right locator position a little earlier than required: you will need to experiment to find the necessary value, which changes according to the song tempo.

Notator/Creator's »Drop-merge« (»Spacebar«) functions as before, allowing a MIDI recording to take place at the same time as a tape recording (ie recording analogue and digital simultaneously!).

8.4 Remote control

In principle, every Fostex function can be controlled via SysEx data. The RMG's freely-definable control elements can be used by advanced MIDI SysEx users to configure their own user-interface: the required MIDI information is obtainable from Fostex.

In this context, the MIDIA program is very helpful: its Byte Calculator can translate hexadecimal and binary code and create SysEx messages which can then be tested on the MTC-1. SysEx files can then be saved as a track (».SEQ« file) for loading into Notator/Creator.

Putting tracks into Record Ready, activating the »Input Monitor« mode and much more can be remotely controlled from the RMG's switches and faders. Especially critical punch procedures can be preprogrammed by entering the corresponding P_USER events in the event list.

You may even »remote control the remote control« by using Realtime Transform to, say, select the track number via Program Changes, drop in/out via a sustain pedal, etc.

Chapter 27

Hardware Peripherals: Unitor, Human Touch, Steady Eye, Export

27

1. Unitor: the basics

The capabilities of Creator and Notator are vastly enhanced by the addition of dedicated hardware:

Unitor C for Creator
Unitor N for Notator.

Unitor combines four distinct functions:

1. SMPTE/EBU synchronization
2. MIDI communication
3. Multiport for connecting Human Touch or Steady Eye
4. Built-in dongle

The SMPTE synchronization is made possible by SMPTE timecode. Because of Creator/Notator's extremely high tempo resolution, the length of a piece of music can be worked out to the nearest few milliseconds, allowing, for instance, the simulation of timing characteristics of other synchronizers.

TWO ADDITIONAL MIDI OUT PORTS (»E« and »F«), together with the ST's existing MIDI Output (»A«), treble the number of MIDI Channels and the speed of transmission. Adding Export to the system gives six MIDI Outputs with a total of 96 MIDI Channels.

TWO ADDITIONAL MIDI IN PORTS (»II« and »III«), together with the ST's existing MIDI Input (»I«), allow the MERGING of three distinct sources of MIDI data simultaneously.

There are no controls on Unitor: all commands are in Creator/Notator.

Never insert or remove Unitor without first switching the computer off.

2. MIDI Outputs E and F

Unitor has two additional, individually independent MIDI Outputs: »E« and »F«.

In Creator/Notator, each track's »CHANNEL« parameter allows the selection of an Output. Because each of the six Outputs has sixteen MIDI Channels, adding Unitor and Export increases the available Channels to 96. Without Export *(see section 16)*, there are 48 Channels over Outputs A, E and F *(see also Chapter 6 »Tracks«, section 3, where the advice also applies to Outputs B to F)*.

Even if there are more than enough Channels, the MIDI devices should be spread out over the available Outputs for timing reasons. Timing is improved when data is transmitted in parallel over different ports.

If Unitor and Export are in use, »critical« instruments (used for drum sounds) and heavy MIDI data loads should be transmitted over Outputs A, B, E and F.

In this configuration, Outputs C and D on Export should be used for single instruments only.

3. MIDI Inputs II and III

Unitor has two additional MIDI Inputs: »II« and »III«. Input »I« is the ST's existing input.

MIDI signals are filtered and »realtime Transformed« before being MERGED into the record track and then re-transmitted according to the MIDI Thru function and realtime ghost tracks *(see Chapter 23 »Realtime MIDI Functions«)*.

4. What are SMPTE, EBU, Frame rate, frame, bit?

»SMPTE« is an acronym for »Society of Motion Picture and Television Engineers«: they perfected this timecode for use in synchronizing sound to picture in the USA film industry.

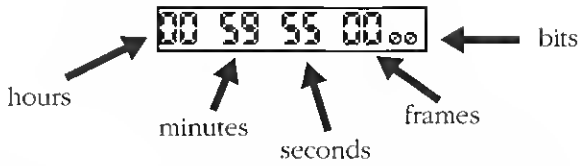
This code is now an audio-industry standard: SMPTE timecode is used in almost every professional context, even outside the film industry.

4.1 SMPTE/EBU timecode format

SMPTE contains information about the exact »time-of-day« position of (in our case) the tape at the tape machine's replay head:

hours – minutes – seconds – frames – bits.

Creator /Notator display this »time-of-day« as follows:



SMPTE's hours, minutes and seconds are self-explanatory: it is a 24-hour clock, each hour is 60 minutes, each minute is 60 seconds.

»Frames« are divisions of a second.

A frame consists of 80 »bits«. One bit corresponds to 500 microseconds (1/2 millisecond) at a frame rate of 25 frames/sec.

The bit is the smallest unit of time in the whole system; the SMPTE OFFSET (when the music starts) can be shifted in these units.

4.2 Frame-rate

The »frame-rate« defines the number of (picture) frames per second.

There are differing frame-rates around the world, even within each branch of the entertainment industry.

American monochrome television has adopted 30 fps (»frames-per-second«). When colour television was introduced, the picture frequency was set at 29.9 Hz, which caused a lot of difficulties when synchronizing; the »30 Drop Frame« standard was introduced to overcome them, by leaving out a frame every so often. All the other frame-rates are »non-drop«.

In Europe, the frame-rate standard is 25 fps, both for monochrome and colour television. Strictly-speaking, the 25 fps timecode should be called »EBU« instead of »SMPTE«, since it was adopted by the European Broadcasting Union, but everyone tends to call all four codes »SMPTE«.

Lastly, the film industry uses 24 fps.

4.3 The various frame-rates

Frame rate	Frames/sec	One Frame [ms]	One Bit [micro-sec]	Bits/sec [Hz]
24	24	41.67	520.8	1920
25	25	40.00	500.0	2000
30 Drop	29.97	33.36	417.1	2400
30 Non-Drop	30	33.33	416.7	2400

5. Choosing the frame-rate

If Unitor is being used purely for audio production, use a 25 fps frame-rate in Europe (30 fps in USA), though strictly-speaking it does not matter so long as Unitor is set to the tape's fps.

In video/film production, it is important to know the frame-rate of the medium being used, and to install it in the »Sync Reference« window.

It is vital that all the tape in a production should have the same frame-rate so that all the machines lock together.

Important: the frame-rate set in the Sync Reference must match the frame-rate being read off the master. Unitor cannot function correctly if the two rates do not match.

The »SMPTE« box below the pattern window will show regular speckled beams if the wrong frame-rate is set in the »Sync Reference«.

6. Basics of SMPTE sync, »Sync Reference«, SMPTE Offset

A track of the audio multitrack is reserved for the SMPTE timecode.

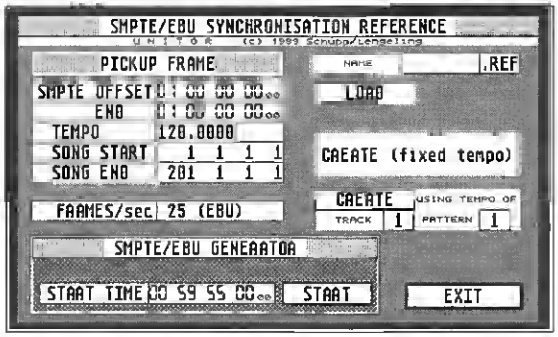
Unitor's job is to establish a clear and solid relationship between the position of the tape and the sequencer's bar position, and to maintain it as accurately as possible.

The information which does this job is contained in the program's »Sync Reference«, which includes:

- the timecode frame-rate
- the SMPTE Offset (the tape's position relative to where the song starts)
- the song's tempo, or changes of tempo.

Almost all synchronization settings are carried out in the »SMPTE/EBU Synchronization Reference« window (»Sync Reference« for short).

Open this window via »Synchronization« (»Options« menu), clicking the »SMPTE« box below the pattern window or pressing ‹S›.



A Sync Reference, once »created«, forms part of a Creator/Notator song (use »Load/Save SONG« in the »File« menu). It can also be saved/loaded separately *(see section 11)*.

Unitor reads the SMPTE code coming from the tape. According to the current tape position and the »Sync Reference« data, the song's exact position is calculated and the music played.

Having SMPTE information in absolute units of time combined with a »Sync Reference« has three distinct advantages compared to other sync methods:

1. Ease of use: you can start synchronizing wherever you like, even halfway through the music.
2. Reliability: mistakes such as drop-outs in the timecode can be recognized and corrected »on-the-fly«.
3. Editing: by altering the SMPTE Offset, the music in the sequencer can be moved in time against the master tape even after recording, and can be lengthened or shortened in time by altering the tempo.

Furthermore, menu »Position in Frames« (»Flags« menu) displays all time positions in terms of the SMPTE tape position *(see Chapter 5 »Positioning«, section 1.4.2 »Position in frames«)*.

This allows you to exactly match music/effects to a video cue *(see Chapter 9 »Editing in the Event Editor«, section 5.1.3 »Pickup clock«)*.

7. Generating SMPTE/EBU timecode

Recording (»striping«) timecode onto a multitrack tape or cassette should be done before any music is played on it; it is best to stripe the whole tape in one go. Use one of the edge tracks (normally the highest one) to minimize cross-talk. Do not use the neighbouring track for percussive or bass-heavy sounds. This may corrupt the timecode due to cross-talk.

Connect Unitor's »SMPTE OUT« 1/4-inch jack socket to the tape machine's track input. Unitor's output level is set at −6 dBm (0 dBm = 0.775 V). Unitor can deliver balanced (stereo jack) as well as unbalanced (mono jack) signals.

The recommended level to tape is between −10 to −6 dB VU on professional machines, and −3 to 0 dB VU on cassette machines. Switch out any noise reduction (especially »dbx«) on the timecode track before recording.

To generate the timecode, set the frame-rate (»FRAMES/sec« box) and the »GENERATOR START TIME« in the »Sync Reference« window, then click »START«. Press any key to end the striping.

There is no set rule as to what the start time of the timecode on tape should be, but most studios start the timecode at 00:00:00:00 (hour : minute : second : frame), then make the Offset 00:00:20:00 at which the sequencer starts.

You could also start generating at 00:59:55:00, and make the Offset five seconds later at 00:01:00:00, which is a nice round value.

Do not generate timecode starting after around 23:00:00:00, since »midnight« causes problems. Apart from that, the time of the timecode on tape is relatively unimportant since the Offset in the »Sync Reference« can start the music start at any time.

Important: you must generate the correct frame-rate: Unitor cannot subsequently read it at a different rate. If in doubt with audio to audio, choose 25 fps in Europe and 30 fps in America.

7.1 Phase-synchronous timecode

Using Unitor (without Steady Eye) in video work, the SMPTE timecode on the video sound track is not »in phase« with the picture. This applies to all other similar SMPTE synchronizers. In other words, a »frame« of the timecode is not necessarily at exactly the same position as a video »frame«.

The question whether the music and video are locked frame-to-frame is important if at some point there will be some electronic editing or »laying-in« of the music to video: if sound and picture are phase-locked, no problems.

To be able to generate SMPTE timecode synchronized, frame-to-frame, to the video picture, Unitor with Steady Eye can read and analyze the video signal itself. This needs only doing once, in order to generate the synchronized SMPTE timecode which is then recorded directly onto the video's audio track.

What happens is that Steady Eye reads the incoming video's »frame edge« information and locks the outgoing SMPTE timecode's frames to them. It is not receiving any »time« information, only locking information.

(In addition, Steady Eye can read »VITC« timecode, but this is unrelated to the above feature. »VITC« timecode must be already inserted on a video for Steady Eye to be able to read it, but it allows Creator/Notator to remain locked to the picture even in freeze frame and slow-motion play. VITC, by definition, frame locks the program to the video with the advantage of dictating the »time-of-day«. Please consult the Steady Eye manual.)

8. SMPTE synchronization

8.1 »Creating« a »Sync Reference« without tempo changes

What to do:

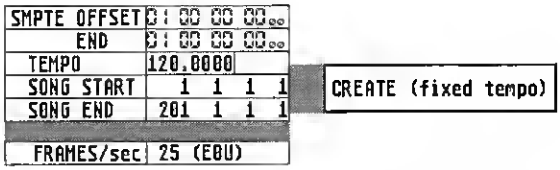
1. There must be SMPTE timecode on tape.
   (If there is none, read section 7.)
2. Connect the tape machine's SMPTE track output to Unitor's »SMPTE« Input.

Unitor's input will work with balanced (stereo jack) and unbalanced (mono jack) signals at between –20 to +6 dBm. It is best to take the timecode straight from the tape machine without going through the mixing desk to get as clean a signal as possible.

To check that Unitor is receiving timecode, switch to »SMPTE Sync« mode and ensure the correct frame rate is set in the »Sync Reference« window: when the tape is rolling,

you should see the SMPTE time displayed in the »elapsed time« box in the »information bar«. As the »Sync Reference« is missing at this stage, Creator/Notator will not start.

3. Create the »Sync Reference« in the »Sync Reference« window: press »S« and install the following parameters:



»*SMPTE OFFSET*« the SMPTE time at which the sequencer should start. Commercial studios tend to set this twenty seconds after the beginning of the timecode which is on tape, and twenty seconds or more after the end of a previous song (see also section 6).

»*SMPTE END*« this is just a display, it cannot be changed with the mouse. The value shows the SMPTE time equivalent of the »SONG END« value below, and is automatically set when either of the »CREATE ...« boxes is clicked.

»Tempo« this is the main page tempo which will become part of the »Sync Reference«. Alter it to the desired value.

»*SONG START*« determines where in the song the sequencer should start when the SMPTE OFFSET time is reached. This is almost always »1 1 1 1«, though bar positions later or earlier are possible.

»*SONG END*« determines when the song should stop. This bar position is almost never used and should be left at the default 201 bars, unless you need greater length. There is nothing to be gained by making the »SONG END« the same position as, say, the Stop pattern (zero) in the arrange list.

4. Ensure the frame-rate is correct and click the »CREATE (fixed tempo)« box.

Once »CREATE ...« has been clicked, the »PICKUP FRAME« icon now reads »Sync Reference exists«, and two boxes appear in the window:

– »DELETE« allows the »Sync Reference« to be deleted.

– »SAVE« (*see section 11 »Saving and loading a »Sync Reference«).*

»EXIT« returns you to the main page, where you should set »SMPTE Sync« mode on the main page. Done!

Now any start of the tape within the SONG START and SONG END positions will start Creator/Notator at the correct position, which will run very accurately with the timecode.

If nothing happens, check that Creator is receiving timecode by looking at the »elapsed time« display in the »information bar«: if it is not, check all your hardware connections; if it is, check that the »SMPTE OFFSET« value is correct in relationship with the timecode coming in.

All the tracks on the tape (except for the sync track) can now be used. You may find it useful, even if you will always be running the program in sync with the tape while you record any voices or acoustic instruments, to make a rough submix of the MIDI instruments onto one track of the tape so that the program does not have to be there all the time while you record non-MIDI stuff.

Always record your music *after* the SMPTE timecode is recorded. Having to synchronize to music that was recorded without a sync track is possible, but best avoided if possible.

Tip: even when the program is stopped, while in »SMPTE Sync« mode it must always be watching its SMPTE Input for signs of timecode. If you are not using SMPTE for some time, it makes sense to switch into »Internal Sync« mode, which puts the program into its fastest mode of working.

8.2 »Creating« a »Sync Reference« with tempo changes

There are two ways of creating a »Sync Reference« with tempo changes: if all the tempo changes *(see Chapter 25 »Tempo«)* are contained in one track, use the following method. If tempo changes are spread around different tracks and patterns, use »Learn SMPTE/Internal« *(see section 10.3)*.

Please read the above section on creating a »Sync Reference« without tempo changes, which applies here apart from the tempo and »CREATE« information:

1. There must be SMPTE timecode on tape.
2. Connect the tape machine's SMPTE track output to Unitor's »SMPTE« Input.
3. Create a »Sync Reference« with tempo changes: place the sequencer's track cursor on the tempo track, press »S« to open the »Sync Reference« window and install the »Sync Reference« parameters *(see above section)*.

The start tempo is automatically set and will not normally need adjusting. The »Using Tempo Of« box will default to showing the track on which you have left the track cursor.

4. Ensure the frame-rate is correct and click the »CREATE using tempo of track xx pattern xx« box.

»EXIT« returns you to the main page, where you should set »SMPTE Sync« mode on the main page. Done!

If you subsequently make any changes to the tempo track, you must click »CREATE using tempo of…« again.

9. Editing a »Sync Reference«

Even during synchronization, the SMPTE Offset and Tempo (to the nearest fourth decimal place, eg 120.0001) can be changed in the »Sync Reference« window »on the fly«.

As a result, alterations are very easy. They have an immediate effect (approx 500 ms) on the synchronization thanks to the program's multitasking capabilities, and there are no timing glitches.

Careful! If a sequence has tempo changes, changing the tempo value in the »Sync Reference« will disable them.

You should not alter a »Sync Reference« without good reason. Ensuring that you start with the correct »Sync Reference« can save you a lot of extra work.

There are situations which will force you to make changes to a »Sync Reference«, or even to create one from scratch. The most basic example of this is where you have music on tape whose tempo and start time relative to SMPTE is unknown. The program must be synchronized to this music as closely as possible:

1. If the existing music was created using a synchronizer other than Unitor, the »Learn SMPTE/MIDI« mode is a very convenient way of transferring the other synchronizer's synchronization information to Unitor via MIDI *(see section 10.2)*.
2. If the existing music was created without using any form of synchronization, the »Learn SMPTE/TAP« *(see section 10.1)*mode is a convenient way of dealing with this, by definition, difficult situation. This situation also arises when the music was created using a synchronizer, but you do not know which type, or where there is no parallel click track recorded.
3. If the existing music has a constant tempo with no tempo changes, there is a third way of synchronizing the program:

– There must be SMPTE timecode on tape.

– Set the »SMPTE OFFSET« as accurately as possible in the »Sync Reference« window: to do this, set the program to »SMPTE« Sync mode and enter the window. Play the tape from before the music's start (it helps if there is a count-in on tape) and click »PICKUP FRAME« as soon as you hear the beginning of the song. This will enter the current SMPTE time as the SMPTE Offset value.

– In the »Sync Reference« window, enter the tempo as accurately as possible. This can easiest be done using the »Tempo Interpreter«'s »Tap Count-In«, eg »Tap Step«: 1/4, »Tap Count-In«: 4, so that your four initial taps (plus one) on the MIDI keyboard follow the tempo of the piece you are listening to (»Manual Sync« mode). The tempo is automatically loaded into the »Sync Reference« window.

– Ensure that the »SONG END« setting is generous enough and click »CREATE (fixed tempo)«.

– Set »SMPTE Sync« mode on the main page.

At this point the sequencer should be more or less in time with the music on tape.

– Now »fine-tune« your »Sync Reference« by adjusting both SMPTE Offset and Tempo. You will need tracks on tape and in the sequencer that can be compared, ideally containing kick drums or cowbells etc.

Adjust the SMPTE Offset value at the beginning of the song, and the Tempo towards the end of the song: this automatically optimizes the timing in the middle. Repeat both adjustments until you are satisfied with the result.

At this point, the reasoning behind the tempo's four decimal places becomes clear. It allows a 30-minute piece of music to be adjusted a few milliseconds either way, which is why you adjust tempo by listening to the end of the music.

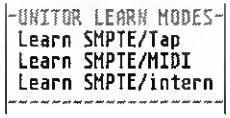
The individual units of tempo and SMPTE Offset (including bits) can be mouse clicked in the usual way. Use of the ›Control‹ & mouse feature is recommended for jumps between values *(see Chapter 3 »Concepts«, section 1.2)* where in-between values are not touched.

While editing (even tempo) in the »Sync Reference« window, the SMPTE time will be shown in the »information bar«.

Alternatively, instead of doing countless adjustments, the »Sync Reference« can be deleted (click the »DELETE« box), new parameters installed, and »CREATE ...« clicked again.

10. »Learning« a »Sync Reference«

The »Learn« modes automatically analyze or »learn« incoming tempo characteristics and create a »Sync Reference« accordingly.



10.1 »Learn SMPTE/Tap«

The »Tempo Interpreter« *(see Chapter 26 »Synchronization«, section 6)* itself reads tempo data from the timing of MIDI notes or from audio/click signals via Human Touch *(see section 14)*.

Example 1: to synchronize the program to an existing piece of music, manually tap to the beat in 1/4 notes:

– Connect any MIDI device (keyboard, drum pad, etc) to one of the MIDI Inputs.

– Switch to »Manual Sync« mode and enter suitable parameters in the »Tempo Interpreter«, eg:

»MIDI NOTES«
»Tap Step« – 1/4
»Window« – 12
»Tempo Response – 1

Now incoming MIDI notes will control the tempo.

– Connect the tape's SMPTE Output to Unitor's SMPTE Input.

– Reset the Main Bar Counter to »1 1 1 1« and click »Learn SMPTE/TAP« (»Options« menu).

– Start the tape and tap your instrument on the beat.

At the end of the music press »STOP«: the »Sync Reference« is now created.

Example 2: to synchronize the program to an existing piece of music, use the available drum track (or better still, a click track) to send tempo triggers via Human Touch. This is much more accurate than playing by hand (as in example 1 above):

– Connect the drum track to a Human Touch input.

– Switch to »Manual Sync« mode and enter suitable parameters in the »Tempo Interpreter«, eg:

»MULTIPORT«
»Tap Step« – 1/4
»Window« – 12
»Tempo Response – 1

Now incoming audio signals will control the tempo.

Where the incoming audio signal is complicated, the setting of the »Tempo Interpreter« parameters is critical, especially as concerns the initial tempo of the music. The tempo shown in »Manual Sync« mode can vary slightly from that shown in the »Learn SMPTE/Tap« mode, though this has no bearing on the precision *(see remarks in section 10.2)*. The audio signals used should ideally consist of regular percussive sounds from a single instrument. However, even mixed-down tracks of rhythmical music can be equally suitable since the drum track normally stands out reasonably clearly from the accompanying music. Carefully trimming the audio input can help, as can use of compression, limiting, equalization etc.

Much simpler than this is the reading of a click track, where small drop-outs will be bridged over.

– Connect the tape's SMPTE Output to Unitor's SMPTE Input.

– Reset the Main Bar Counter to »1 1 1 1« and click »Learn SMPTE/TAP« (»Options« menu).

– Start the tape; the first audio signal to reach Human Touch will start the sequencer. Play the tape all the way through and continue beyond the end if possible to allow the new production to go on longer if necessary. If there is a break in the incoming taps or other timing problems, the program will interpret this as an end to the »Sync Reference« and stop: you are given the choice, via a dialog box, of keeping or cancelling the aborted »Sync Reference«. Click cancel, and also check in the »Sync Reference« window that no »Sync Reference« has been created: if so, click »DELETE«.

At the end of the music press »STOP«: the »Sync Reference« is now created. Save the song to save the »Sync Reference«.

10.2 »Learn SMPTE/MIDI«

With »Learn SMPTE/MIDI«, a »Sync Reference« is automatically created when tempo information from a MIDI Input (in the form of MIDI Clock pulses) is received at the same time as SMPTE timecode via Unitor.

This mode allows the transfer of timing characteristics of a different synchronizer into Unitor without the need for additional commands.

This is useful where the music was originally created using a different synchronizer, and you have access to the same model of synchronizer (though now, of course, it's Unitor you want to use).

- Set up the other synchronizer and Creator/Notator ready for synchronization as usual.
- Connect the synchronizer's MIDI Output to one of Unitor's MIDI Inputs.
- Connect the tape's SMPTE output to both Unitor's and the synchronizer's SMPTE Inputs, so they receive the same time code (assuming the synchronizer reads SMPTE). Avoid using eq etc if you have to use a mixing desk to split the signal.
- Wind the tape back to before the song start, select »Learn SMPTE/MIDI« (»Options« menu) and start the tape.

The MIDI synchronization gets under way as usual: both units are reading SMPTE, but Unitor is reading the Clock pulses from the other synchronizer as well to create a »Sync Reference«.

When finished, switch the program to »SMPTE Sync« mode and put away the other synchronizer: one run-through was enough for Unitor to simulate its timing characteristics for that song.

Note: after the above process, synchronization takes place »one generation removed« from the original: the precision of the synchronization is somewhat affected. A slightly »warbling« tempo display, which can also differ slightly from the other synchronizer's display, does not adversely affect the precision of the synchronization; to optimize the synchronizing, these continuous adjustments are desirable. To achieve the most precise synchronization, Unitor constantly corrects the tempo while reading the SMPTE timecode if there are fluctuations in the tape speed, without altering the »normal« tempo display.

10.3 »Learn SMPTE/Internal«

There are two ways of creating a »Sync Reference« with tempo changes: if all the tempo changes *(see Chapter 25 »Tempo«)* are contained in one track, use the quick method »CREATE using tempo of ...« *(see section 8.2)*.

If tempo changes are spread around different tracks and patterns, use »Learn SMPTE/Internal«.

– There must be SMPTE timecode on tape. Connect the Output of the SMPTE signal from tape to Unitor's SMPTE Input.

– Enter any »SONG START« position in the »Sync Reference« window (usually »1 1 1 1«). Any existing »Sync Reference« should be deleted to avoid unexpected automatic starts. Check the frame rate.

– Switch to »SMPTE Sync« mode and start the tape. The SMPTE readout in the »information bar« will start to roll.

– When you are ready, press ›Shift-W‹ (»Recording Sync Reference«) or click »Learn SMPTE/Internal« (»Options« menu).

This will start the sequencer, which will run as if in normal »Internal Sync« mode with all the tempo changes. At the same time, the »Sync Reference« is being created. The SMPTE Offset time in the »Sync Reference« window will be determined hy when you pressed ›Shift-W‹. Afterwards, you can round this off to the nearest second etc, but do not forget to reset the »SONG START« value to »1 1 1 1« as well.

On the other hand, you can pre-select a SMPTE Offset time and press ›Shift-W‹ *before you start the tape*, which will put the sequencer into »Waiting for AUTOSTART« mode; it will start automatically on reaching the SMPTE Offset time.

You can even alter the tempo (eg with »mouse as slider«) while you are recording the »Sync Reference« – these tempo changes will be directly entered into the »Sync Reference«. If you do this, and you already have tempo changes recorded in the song as well, make sure there are no tempo-change conflicts!

– At the end of the song, stop the tape.

The »Sync Reference« now contains all the tempo changes.

Note: if you subsequently make any changes to the tempo, you will need to start the above process all over again. This is why it is preferable, where possible, to contain all your tempo changes within one track and to use the simpler process as described above in section 8.2 »CREATE using tempo of…«.

11. Saving and loading a »Sync Reference«

A created »Sync Reference« forms part of a Creator/Notator song and will be saved to and loaded from disk via »Save/Load SONG«.

A »Sync Reference« can also be named and saved to/loaded from disk separately via »LOAD/SAVE« in the »Sync Reference« window. The file extension is always »*.REF«.

If you type in a name that is already on disk, the program will ask you if the old file should be renamed to »*.BEF« before saving the current one.

12. SMPTE troubleshooting and tips

Unitor is able to deal with certain timecode problems. Small problems will have no effect on the timing, slightly bigger ones may lead to a slight and very short variation in timing. Very large problems, such as long drop-outs, can lead to the synchronization being aborted. The condition of the SMPTE timecode can be seen by a »beam«-type display in the »SMPTE« box below the pattern window. A good timecode track displays no beam. Small blips are still fine. A large dropout prompts the warning message »Non-contiguous Timecode!«; this will happen only once, after that, it assumes you have heeded the warning. This happens rarely.

To ensure you minimize the error factor, always use good-quality recording tape, ensure the tape machine is aligned and de-magnetized, record and replay at the recommended levels and minimize crosstalk between tape tracks.

Unitor data (0 dBm = 775 mV):

Output: −6 dBm
Input: −20 to +6 dBm /10 kOhm

Recommended recording levels:

Studio multitracks 2" and 1": −10 dB to −6 dB VU
Multitracks 1/2" and 1/4": −6 dB to 0 dB VU
Cassettes: −3 dB to 0 dB VU

You should find out for yourself what the optimum record and playback levels are on your tape machine and on which track the timecode is happiest.

The timecode track is normally recorded on the highest-numbered track. Badly-maintained tape machines can damage edge tracks, so other tracks have to be used for the timecode. Beware, then, of cross-talk, particularly from percussion and bass tracks; the timecode could also bleed to other tracks. The best way is to leave neighbouring track(s) empty.

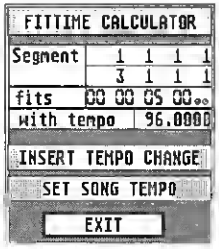
If a synchronizer is connected to a multitrack with an unbalanced cable, it could lead to earthing between the digital MIDI communication and the analogue audio-paths. This can be heard as a slight crackling when, for instance, the modulation wheel is turned.

To remedy this, use a balanced jack cable (Unitor's Input and Output are both balanced and unbalanced).

If the multitrack's Output is unbalanced, instead of using a DI (Direct Injection) box, try the following: at the Unitor end, solder the earth to the middle ring of a stereo jack plug and the signal wire to the tip, leaving the plug's earth connection free.

This simple balancing operation works only if all equipment (Atari, multitrack etc) is properly earthed.

13. »Fittime Calculator«



The »Fittime Calculator« (»Options« menu) helps answer questions that often crop up when laying sound to picture, such as: »what tempo do I need if a whole 4/4 bar is to last exactly two seconds and seventeen frames?«. You may enter normal musical segments (in bars, beats etc), absolute lengths (in hours, mins, secs, frames and bits) or tempo, and receive the appropriate answer.

»INSERT Tempo CHANGE« inserts the required tempo into the current track at the beginning of the current segment of music, after which the »Segment« display automatically scrolls forwards to help you insert a logical sequence of tempo changes. If these tempo changes are to form part of a Unitor »Sync Reference«, follow the method described in *section 8.2*.

»SET SONG Tempo« installs the current Fittime tempo as the main program start tempo.

14. Human Touch

Human Touch serves two main purposes: to control the tempo of Creator and Notator by reading audio signals, and to synchronize external devices via click signals.

14.1 Generating signals

As a method of synchronization, the use of click signals is now out-of-date: its operation is more complicated, and is far less user-friendly (eg always having to start at the top of the timecode) than the later MIDI Clocks with »Song Position Pointers«. However, synchronization click tracks can still be a necessity with some older devices.

14.1.1 Click Outputs

Human Touch has three independent Outputs which can each give out differing click signals. A single »click« consists of a rising and a falling edge.

A »P_USER 21« event will transmit a single click at any time position, allowing you to assemble any type of click track at any tempo, even with pauses.

P_USER 21's »second data byte« (as seen in the event list's »–2–« column) determines the Output as follows:

HUMAN TOUCH Output	VALUE –2–
1	1
2	2
3	4

The same P_USER 21 click can be sent to more than one Output by adding together the respective P_USER values (1, 2, 4):

HUMAN TOUCH Output	VALUE –2–	
1 2	3	(1+2)
1 3	5	(1+4)
2 3	6	(2+4)
1 2 3	7	(1+2+4)

Eg: *1 1 1 1 P_USER 21 1 HuTouch Click* means a click is sent to Output 1 at the start of Bar 1.

To assemble a click track for sync purposes, simply mouse-enter a few such P_USER events into a track at the desired time positions (using »Segment Copy« can help) and use the »LOOP« track parameter to make up a continuous track.

Example: for 24 clicks-per-quarter note, enter 24 »P_USER 21« events, each eight 1/768th notes apart, within the first quarter note:

1 1 1 1
1 1 1 9
1 1 1 17
1 1 1 25, etc

then set »1« (one beat) as the »LOOP« track parameter on the main page. (This particular 24-click rate can be created even more simply using »P_USER 25« – *see below.*)

Tip: because each click position can be precisely set, swing rhythms can be achieved by the deliberate delaying of every second 1/8th note *(see Chapter 17 »Quantize, ...«, section 2 »Groove Design«)*.

A P_USER's click signal is quite short (sawtooth); some drum machines require a squarewave instead: *see next section on »P_USERs 22/23«*.

The most common use for a click-type of synchronization is to slave older drum machines that use the 24-clicks-per-quarter note »Roland DIN Sync«. »P_USER 25« is programmed to produce a 24-click stream:

P_USER 25 1 Switch DIN switches DIN Sync on
P_USER 25 0 Switch DIN switches DIN Sync off

These DIN Sync signals will always appear at Output 2. Even here, though, it can be necessary to create a square-wave signal instead of these sawtooth ones *(see next section on »P_USERs 22/23«)*.

Many DIN Sync devices additionally require a Start/Stop signal: when using the P_USER 25 DIN Sync signal at Output 2, Human Touch gives out a Start/Stop signal at Output 3; in Play mode, this Output goes HIGH, in Stop mode LOW.

During DIN Sync, Output 1 remains free for other uses.

14.1.2 Switching/control Outputs, squarewave signal

Each of Human Touch's three Outputs can be separately programmed with P_USER 22 and 23 events. To be able to use these properly, a minimum of technical know-how is useful.

A P_USER 22 event will switch an Output to a 5 Volt HIGH. This signal *remains* high (unlike a »click« signal: see above) until a P_USER 23 event switches it back to a 0 Volt LOW.

P_USER 22/23's »second data bytes« determine the Output. The same P_USER 22/23 signals can be sent to more than one Output by adding together the respective P_USER values *(see section above)*.

Example:

P_USER 22 1 HuTouch HIGH switches Output 1 to HIGH
P_USER 23 1 HuTouch LOW switches Output 1 to LOW

Entering a sequence of these P_USER 22/23 events allows the creation of a switching/controlling signal for any purpose.

To create a 50% duty cycle squarewave signal, the P_USER events must be regular:

```
1  1  1  1   P_USER  22  1   HuTouch HIGH
1  1  1  5   P_USER  23  1   HuTouch LOW
1  1  1  9   P_USER  22  1   HuTouch HIGH
1  1  1  13  P_USER  23  1   HuTouch LOW
1  1  1  17  P_USER  22  1   HuTouch HIGH
1  1  1  21  P_USER  23  1   HuTouch LOW  ... etc
```

This creates a 24 clicks-per-quarter note signal (one squarewave every eight 1/768th notes). If DIN Sync is also switched on, there is a Start/Stop signal at Output 3 if required.

14.2 Click and audio Inputs

Human Touch has three Inputs intended for click and other audio signals, such as a kickdrum microphone.

The lefthand Input switch on top of Human Touch switches between the Inputs and the built-in microphone. Microphones and other low-signal sources go into the sensitive Input (»MIC«). Very high-level signals such as from a tape machine go into the second Input (»LINE«). »AUX« is reserved for future projects.

The righthand switch acts as a high/low *pass* filter and selects between frequency bands. If the acoustic source is, say, a hihat, you can filter out the unwanted lower frequencies which might cause false triggers by switching to »high«. The switch is usually left in its »flat« (linear) position.

The rotary sensitivity control regulates the sensitivity of the built-in microphone, the mic and the line Inputs.

The primary uses for Human Touch's Inputs are in the fields of post-production (post-synchronization to music on tape) and in the control of tempo in live performances (eg by a drummer).

In the »Tempo Interpreter« window, there are two boxes that deal specifically with Human Touch signals:

MULTIPORT allows/disallows Human Touch acoustic signals to be accepted as tempo triggers.

MP-DISCRIMINATION has a »masking« action: it shows the number of milliseconds that have to pass after an acoustic signal before the »Tempo Interpreter« will accept a new one, ie signals that follow on too quickly will be ignored.

This parameter is very important if you are trying to extract tempo information from acoustic signals (as opposed to pure click signals – see below). Sounds consist of many consecutive vibrations: the necessary information for the »Tempo Interpreter« lies in the *attack* portion of a sound, but not in each »vibration peak« within the ensuing sound envelope. The Multiport Discrimination function ignores this additional series of vibrations, accepting only the *first* such vibration of a series. A new sound will be accepted as tempo trigger only after a »pause« (of, say, 60ms) following the previous sound. Even percussive sounds perceived by us as being »short« are actually long enough to need the discrimination treatment.

The »DISCRIMINATION« setting should therefore be set long enough to avoid »multiple triggering«, eg 40 to 100 ms.

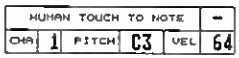
With pure click signals at the Input, this discrimination must be shorter than the distance between the clicks, eg 0 to 15 ms.

If signals are accepted by the »DISCRIMINATION« function, they are passed on to the »Tempo Interpreter« for further processing *(see Chapter 26 »Synchronization«, section 6)*.

14.3 Creating MIDI notes via Human Touch

Trigger signals at any of Human Touch's Inputs can be converted into a MIDI note in realtime. In the »Tempo Interpreter«, set the MIDI Channel, pitch and velocity and switch »Human Touch TO NOTE« on.

This allows the created notes to be manipulated in the program as if they had come in via one of the normal MIDI Inputs.



»DISCRIMINATION« is still operative in this mode.

Uses: replacing an existing kickdrum on tape with a sampled one; or creating a »Groove Design« template track from an acoustic signal (eg kickdrum track) etc.

15. Steady Eye

15.1 Overview

Steady Eye fulfills two video synchronization-related tasks, together with Unitor and Notator.

1. It reads »VITC« from a video signal and locks the system to the video;

2. It produces a frame-locked (»phase locked«) »LTC« audio signal as a result of reading a video signal.

Each of these two distinct operations enables frame-synchronous working to take place where video is the tape medium. It is vital to understand what is meant by »synchronization«, »SMPTE/EBU«, »LTC«, »VITC«, and »frame- (or »phase-«) locked«.

15.2 What are LTC and VITC?

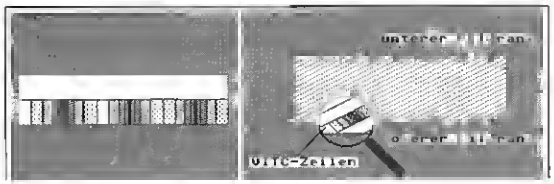
SMPTE/EBU timecode contains the information about the exact time of a tape position, and is expressed in hours, minutes, seconds, frames and bits (*see Chapter 27.4 »What are SMPTE, EBU…«).*

There are two ways to produce SMPTE/EBU timecode:

1. As an AUDIO signal: LTC (Longitudinal Timecode)
2. As a VIDEO signal: VITC (Vertical Interval Timecode)

SMPTE/EBU LTC is widely-used: the timecode appears as an audio signal which is recorded onto a multitrack tape or onto a video's audio track (normal or »HiFi«). Virtually all SMPTE/EBU synchronization devices work this way, Unitor included. Because of this, the additional description »LTC« tends to be left out.

SMPTE/EBU VITC has traditionally only been used in the world of professional video production: the timecode forms an invisible part of the video picture, and is therefore an integral component of the video signal.



SMPTE/EBU-LTC on an audio multitrack

VITC on video

Do not confuse VITC with a »burnt-in« time that you can sometimes see in a corner of the picture. VITC is digital data that is inserted in the picture's invisible lines (numbers 10 to 22). (»Video text« is also inserted here, something you may have already come across in your television.) There is no visible way of knowing whether a tape has been VITC-encoded.

(»Burnt-in« time is a visible SMPTE/EBU clock display that matches the actual time-code, and is inserted into the picture, usually in one corner or along the bottom edge,

so the composer etc can see where he/she is in relation to the timecode. It is inserted there by the tape transfer studio at the same time the LTC timecode is recorded onto the audio tracks, or the VITC is encoded into the picture.)

You are not able to generate VITC yourself, and so cannot record it onto your video: this must be done by a tape-transfer studio, dubbing suite or similar.

Synchronization with VITC, as opposed to LTC, has certain unique properties:

- A video frame has exactly the same position on the tape as a frame of VITC.
- Video frames and VITC frames are therefore inseparably locked together (also called »phase-locked«), and remain so however slowly the tape is played. (For the separate matter of generating phase-locked LTC, see below.)
- If ever the tape is recorded over again with material containing VITC, the VITC is (of course) included since it cannot be separated, so no risk of going out of sync.
- The video's audio tracks remain free and can be used for other purposes.
- VITC can basically be read whenever a frame is visible, which even includes freeze-frame and slow motion search modes (forwards or backwards) if your video tape recorder (or »VTR« as we shall call it) is of good quality and allows these modes.

Regarding the precision of the playback (wow/flutter etc), the use of VITC offers no advantage over that of LTC. The precision depends on the quality of the VTR and of the signal. Whatever the situation, Unitor, Steady Eye and Creator/Notator will attempt to get the best out of your VITC or LTC.

15.3 Basic points on video synchronization; which sync method?

For reasons of quality when putting sound to picture (video, or film which has been transferred to video), the soundtrack is normally recorded on its own audio tape, separate from the video, and is later dubbed onto the video where applicable. When using a sequencer to do the soundtrack, there are certain conditions that must be met:

1. The synchronization must be reproducable and precise. This condition is already met by the use of Creator or Notator with Unitor.
2. The timecode must be phase-locked, ie locked to the picture whatever the tape speed.
3. If the timecode is burnt-into the picture so the user can read it, this must agree with the actual timecode.
4. Synchronization should be possible in freeze-frame and search modes.

With the Steady Eye video synchronizer (together with Unitor and Creator or Notator), all or some of these conditions can be met by using one of two methods:

1. The generating of SMPTE/EBU LTC which is phase-locked to the video frames, and recording this LTC onto the video's audio tracks.

This method has nothing to do with VITC!

This sync method differs from the normal Unitor synchronization, in that LTC (Unitor's SMPTE OUT) is generated as a result of Steady Eye reading the video signal itself: this makes it frame-locked. This is a once-only operation using Steady Eye at the time of generating the LTC; the subsquent reading of this frame-locked LTC is done by Unitor without Steady Eye. Any delay between any (visible) burnt-in time and the LTC you have just recorded is constant, and can be compensated for by use of the »Display Offset« function in the »Sync Reference« window. This method meets the above conditions 1 to 3.

Steady Eye's ability to read the video signal itself is quite separate from its ability to read VITC, and you should not confuse the two. Any video can be »read« by Steady Eye: it detects the frame edges which act as a simple clock, as a result of which it can make Unitor generate LTC that is locked to the frame edges.

This method is the one to use if you cannot obtain VITC for your video tape, or if your VTR cannot cope with VITC. However, your VTR must have the ability to dub audio (ie the LTC) onto its audio tracks without this affecting the picture. Some VTR's cannot do this.

If this is the method for you, you need only read the introductory sections, and the section called »Generating a phase-locked SMPTE/EBU LTC«.

2. The insertion of SMPTE/EBU VITC into the video picture by a dubbing suite, etc.

VITC sync via Steady Eye (with Unitor) guarantees that picture and timecode are locked. This method meets all four above conditions. If the video is played slowly via slow search or frame advance modes, Creator or Notator will react accordingly with the appropriate slow tempo.

This method, unlike the one described above, requires Steady Eye the whole time since it has to continuously read VITC.

This method requires you to read the whole of this manual about VITC, missing out »Generating a phase-locked SMPTE/EBU LTC«.

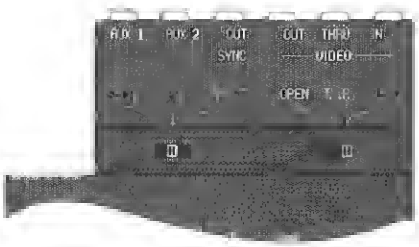
15.4 Connecting Steady Eye, the VTR with monitor

Steady Eye is connected to Unitor's Multiport.

Caution! The computer must first be switched off!

Other peripheral hardware such as the VTR and monitor or television may be connected only after Unitor has been connected to the computer, and Steady Eye has been connected to Unitor. Please never connect or disconnect Steady Eye while the computer is switched on.

Steady Eye's ports:



VIDEO IN – Video input, impedance switchable between 75 Ohm and open, video signal 1 Vpp (Volt Peak-to-Peak)

VIDEO THRU – connect a video monitor here.

VIDEO OUT – (Reserved for future use)

SYNC OUT – Video V-Sync signal for externally sync'able VTRs

AUX 1/2 – (Reserved for future use)

Connect the VTR's video out port to Steady Eye's VIDEO IN, using a standard video cable. If your VTR requires special cables, these are obtainable from your video hardware dealer.

Connect the video monitor to the VIDEO THRU port. The end of a video connection must be of the correct impedance. If no device is connected to VIDEO THRU, Steady Eye becomes the end connection, and so the input impedance must be set to »75 Ohm«. If a device such as a monitor is connected to the VIDEO THRU port, set the impedance to »OPEN«. The third setting, T.I.P, stands for »Try If Problems«....

After connecting the hardware, switch the computer on and start Creator or Notator. As a short test after loading, switch the Creator/Notator »Sync Mode« to »VITC SCAN« or »VITC LOCK«. The first time you switch to one of these two Sync modes, Creator/Notator checks the Steady Eye interface. If there is a problem of non-recognition, the

Alert Box »Steady Eye needed for VITC sync« appears: switch off, count to twenty, then start again. Use the official power switch, not the »warm re-start« button.

15.5 Generating a phase-locked timecode (SMPTE/EBU LTC)

There are two ways for Steady Eye to generate LTC that is locked to picture, frame-by-frame (*read also section 15.3.1*):

1. You can synchronize Unitor's LTC generator to an incoming video signal:

– Connect the VTR's output to Steady Eye's VIDEO IN.

– Connect Unitor's SMPTE Ouput to the VTR's audio input.

– In the »Sync Reference« window, set the generator mode to »Video frame lokked« with the desired start time and the correct frame rate (the frame rate must match the video's one).

– Start the LTC generator and record the LTC onto the video's audio tracks, taking care not to wipe the picture. Your VTR must be able to record audio separate from the picture.

2. If you own a VTR that can be externally-synchronized, it can be sync'ed to the LTC generator:

– Connect Steady Eye's SYNC OUT to the VTR's sync input. Switch the VTR to its external sync mode. (Steady Eye transmits a V-Sync signal at its SYNC OUT port while generating takes place.)

– Connect Unitor's SMPTE Ouput to the VTR's audio input.

– In the »Sync Reference« window, set the generator mode to »Quartz clocked« with the desired start time and the correct frame rate (the frame rate must match the video's one).

– Start the LTC generator and record the LTC onto the video's audio tracks, taking care not to wipe the picture. Your VTR must be able to record audio separate from the picture. In all other details, proceed as directed in Chapter 27.7 »Generating SMPTE/EBU timecode«.

15.6 Inserting VITC - what is involved? Which VTRs to use

There are several factors that affect the quality of VITC sync:

– The picture quality of the cassette
– The VTR
– The VTR's settings and »VITC line«
– Type of VITC encoding

The need to use video cassettes of the highest quality is self-explanatory. Do not use those of dubious quality or age.

So-called »professional« VTRs are the best to use, but also the most expensive. VITC tends to be used with this type of machine only.

The professional's choice is a »U-Matic« VTR. However, the best VHS and Betamax machines can give equally good results for reading VITC. The factors which most influence success in reading VITC are the quality of the VTR, combined with having a tape that has VITC encoded on as many lines as possible (see below).

However, Steady Eye was developed with users' domestic VTRs in mind. Because of the way they are built, these machines deliver unstable video signals, a problem which is hardly noticeable in the picture, but which makes VITC sync difficult.

VTR Criteria:

- Very good quality of freeze frame picture
- Very good quality of slow-motion search mode picture
- As little vertical movement as possible during freeze frame and search modes (both directions) compared with playback at normal speed
- Adjustable tracking, separate for normal play and freeze frame. If your VTR has an »auto« or »digital« tracking setting, disable this, because we need to achieve the best possible quality in the upper part of the screen.
- Any electronic dropout-compensation function should be switched off (often called »EDIT«, »PCM«, etc).
- Adjustable picture sharpness. If there is no such function, an average sharpness is preset. This artificial sharpness is achieved by the conditioning of image details within the VTR's resolution. If you have the option of improving the resolution by using a »sharpness« function, this probably introduces a distorted signal that will impair VITC readability. The amount of sharpness should therefore be set low.
- For the sake of your comfort, transport controls and positioning should be simple and efficient. The ideal is that a VTR should have a »jog/shuttle« wheel to access the desired frame, or at least the equivalent buttons to allow slow motion play forwards/backwards. If you wish to use a remote control device, ensure the frame positioning controls are on it, and that it is quick and reliable.

The use of low-price VTRs cannot be recommended: they are too unreliable for VITC playback. They may well be okay for normal-speed playback, but not for freeze frame or slow motion replay.

Even »better« domestic VTRs do not always deliver a reliable VITC signal: freeze frame/slow motion accuracy tends to be the biggest problem, due the slipping of the whole picture up or down (»vertical hold«). This affects the exact position of the VITC line(s). Steady Eye is limited as to how much it can compensate for these deviations.

One solution to these problems is to encode VITC on more than the two lines normally used: you could have the tape's lines 10 to 22 encoded with the identical VITC signal, giving Steady Eye a better chance of reading the signal even if the picture slips.

You cannot encode the video with VITC yourself: this can only be done in a video dubbing suite or similar. Unfortunately, not all such establishments can carry out this encoding on several lines, but are restricted to the two lines that represent the SMPTE/EBU standard.

With two lines encoded, you are limited as to what you can do if your VTR is of domestic quality:

- the search modes may be unreliable; this depends very much on the VTR's quality
- in freeze frame mode, some VTRs have very poor frame stability. If no valid VITC data is received, Creator/Notator will stop. Again, the VTR's quality is the deciding factor.

So try to obtain multi-line VITC encoding if your VTR cannot handle the above criteria satisfactorily.

Tip: sync accuracy during normal playback is not impaired by unreadable VITC, even with the cheap VTRs, as Creator/Notator can also lock to the frame-edge information of the video signal itself.

15.7 Setting up the VTR, VITC line and VITC timeout

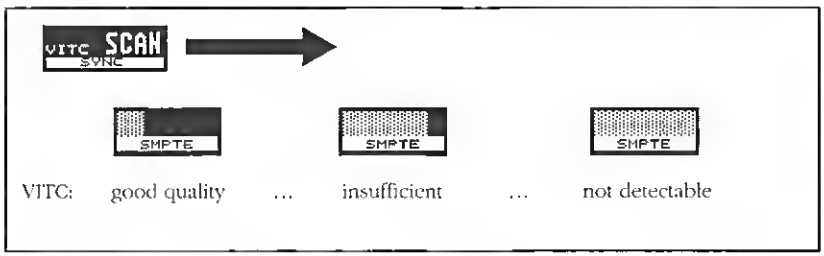
Important! Please follow these settings carefully as they influence the success of Steady Eye's operation.

Set the »SECAM/PAL/NTSC« switch to the correct position, the impedance switch to »OPEN« if you have connected a monitor (otherwise to »75 Ohm«). (Video connections must end with the correct impedance: one device must therefore be connected at the end, whose impedance is 75 Ohm.)

Please ensure the »Sync Reference« window's frame rate is correct. European »PAL« needs a rate of 25 fps.

Set the »Sync Mode« to »VITC SCAN« and observe the SMPTE position display and SMPTE quality display (the »SMPTE« box below the pattern window) as you try each operation of your VTR. No blips across the display is the ideal. Any blip less than

halfway still signifies a relatively good VITC signal; a blip more than halfway across the display indicates a poor VITC reading. If the display shows a full-width reading (grey beam completely visible), this means »not receiving any VITC at all«.



VTRs have electronic switching that suppresses or evens out interference in the picture. This switching can impair the VITC signal. Some VTRs allow the user to disable this feature, and you should do so if possible.

Tracking for normal playback mode and, where available, freeze frame/slow motion modes should be adjusted so as to show minimum disturbance of the picture. If the VTR allows the regulation of flimmer or vibration in freeze frame mode (usually a little control at the back or underneath), adjust it carefully. At all times pay attention to the upper part of the picture, for that is where the VITC signal is inserted. Any automatic tracking function should be switched off if possible.

In the »Sync Reference« window, the default VITC timeout of 100 should be enough (these are half-frames, ie approximately two seconds). Timeout defines how long Creator or Notator will wait following the non-recognition of VITC during VITC sync, before switching to Stop mode. If there is no video signal, Notator stops immediately.

The VITC line setting is best tried while the VTR is in operation, with Creator or Notator in »VITC SCAN« Sync mode. Find out the VITC line value that gives a continuous timecode readout, at the same time giving the fewest blips in the SMPTE quality display.

Where you are using VITC on several lines, many of them should show the same good quality readout, without any difference in quality. The SMPTE quality display should show few blips. Optimize the VTR's settings (tracking, picture sharpness etc) as described above, then check which is the best line to read, especially in freeze frame and slow motion modes.

Tip: tracking cannot normally be adjusted in freeze frame mode, so choose the slowest possible tape speed instead.

With the standard two lines encoded, SMPTE quality will not be changed by trying any of the VITC LINE values, other than within a relatively narrow, sharply-delineated area of the VITC lines (eg 16/18 or 17/19).

Once you have found the best line for all operations, optimize the VTR's settings to get the best SMPTE quality display.

The success or otherwise of these efforts depends very much on the quality of the VTR. You may have to compromise to find the VITC LINE setting that displays the best all-round sync results. Freeze frame and slow motion playback sync is not always possible with two-line encoded tape.

15.8 Synchronizing with VITC

The sync'ing of Creator or Notator to VITC is similar to the sync'ing to LTC. *Read again above sections 7 to 11.* The principles of the »Sync Reference« remain the same, so become familiar with this before embarking on VITC work.

Starting the VTR at the normal speed will cause the SMPTE position display to run. If this position is within the Sync Reference, Creator/Notator will start.

You can select between »VITC SCAN« and »VITC LOCK« Sync modes:

VITC SCAN mode assuming that your VTR and the VITC signal satisfy the quality requirements (see above), the speed and direction of the tape may be changed at will: Notator follows the tape's position, with a slight delay at each change of tape speed/direction. VITC SCAN is intended for editing and note entering, especially with the »PICKUP FRAME« function (/ key in the event editor) and the »MIDI Pickup Frame Input« function (MIDI Step Input within VITC sync) (see below).

VITC LOCK mode Creator or Notator will only work when the tape speed is normal. This allows it to lock more closely to the VITC (as it is also reading the video frame edge pulses) and should therefore be used when the greatest accuracy is demanded. This is the mode to use when recording the final version to audio tape after all the editing has been done. In VITC LOCK mode, there is a longer delay before Creator or Notator start.

If the sequencer stops without warning during VITC sync, it must be because Steady Eye was not able to read the timecode at some point. Check all the settings (see above). Only then should you try increasing the VITC TIMEOUT value in the »Sync Reference« window: this makes the program wait longer before tripping out after bad or missing code. The slight disadvantage is that the program then takes a little longer to stop after an intended tape stop.

If the synchronization does not stop, but the sequencer runs erratically at the tape's normal speed, please check the »Frame/sec« rate in the »Sync Reference« window. PAL tape requires 25 fps.

In some ST computers, especially the Mega ST's, the »blitter« chips can prove problematical for synchronization. Try switching out the blitter on the ST desktop before starting the program; failing that, the blitter chip may have to be removed altogether.

Tip: if you stopped the sequencer during synchronization, simply press the ›Y‹ key twice to restart it in sync with the running tape.

15.9 Helpful synchronization features

15.9.1 Display Offset

The DISPLAY OFFSET parameter in the »Sync Reference« window can be given a value which will be subtracted from all SMPTE displays, except the main SMPTE display next to the Main Bar Counter. For example, if the SMPTE OFFSET value is also entered in the DISPLAY OFFSET readout, events will be shown as starting at »00:00:00:00« (»Position in Frames« switched on).

Likewise, this parameter can enable the program to show the same time as is displayed in any burnt-in time display on the screen. With VITC synchronization, this is not normally necessary as the two ought to agree.

15.9.2 Pickup Frame

The »Pickup Clock« function in the event editor (/ key) allows an event to be given the current time position.

When valid timecode is received (eg VITC), the SMPTE frame position serves as the basis for giving the event a time value. The program rounds the current position to the nearest frame, calculates the bar position (in 768ths or 1536ths) and inserts the event at that position.

15.9.3 MIDI Frame Input

If MIDI Step Input is enabled, and when valid timecode is received (eg VITC), notes entered will be given the exact current time position in frames: to do this, the program rounds the current position to the nearest frame, calculates the bar position (in 768ths or 1536ths) and inserts the event at that position.

Example: in VITC SCAN mode, with MIDI Step Input on, move the tape to a selected frame. Play a key on the MIDI keyboard that triggers the sound that belongs to that

frame. The note is automatically inserted at the current frame position, without your having to do any manual positioning.

16. Export

Export is a »MIDI Expansion Interface« with three additional MIDI Outputs (»B«, »C« and »D«).

Together with port »A« (Atari MIDI Output) there are therefore four separate available MIDI Outputs; if Unitor is added to the system, then six Outputs are separately addressable.

Each individual port has sixteen MIDI Channels. Each port is identified in by a capital letter: port A is the existing, built-in port in the Atari; ports B, C and D are in Export, Unitor has ports E and F.

Each sequencer track can be assigned to one of the six ports. The »CHANNEL« track parameter determines the port (A to F) and its Channel (1 to 16).

Any combination of port/MIDI Channel is possible (up to 96), and all 96 combinations can be given a »Channel name«.

In larger MIDI systems where lots of data is used, Export will further improve timing of events due to its ability to send data in parallel: there are more »openings to the outside« for MIDI data to pass through; this becomes all the more evident the bigger the MIDI system and amount of MIDI data in use.

Several MIDI devices in a system should be connected to the various ports, and one way would be connect your drum machine/sampler containing percussion and drum sounds to port A and your bass device to port B.

If Unitor and Export are used together, »critical« instruments such as those producing drum sounds or large amounts of data should use ports A, B, E and F. In this case, ports C and D in Export should receive only single instruments.

If Export is used without Unitor, timing-critical instruments should be connected to A and B.

System Exclusive messages may be sent through any of the ports, but do bear in mind that when transferring whole sound banks of data, the ports cannot be completely independent one from the other – they have to wait for each other to catch up during these very long messages. You cannot expect port A to keep perfect time if port B is transmitting a whole, say, DX7 soundbank, though the most independent ports are A,

B, E and F. Perfect time should, however, be possible when dealing with single patch data.

See also Chapter 6 »Tracks«, section 3.1.3 »MIDI Out ports«.

Chapter 28
Data Management

28

While you are working with Creator or Notator, all recorded data is temporarily held in the computer's Random Access Memory, or »RAM«. If the computer's power supply is switched off or interrupted even for a split second, the contents of the RAM will be lost forever. You should therefore regularly save the RAM contents to disk for more »permanent« storage.

1. Data organization on disk, »files«

The program's disk operations are in the »File« menu. These concern the saving and loading of your work in the form of »files« in a format recognizable to Creator and Notator. A disk must be »formatted« *(see section 8)* before it can accept files.

A file is given an eight-character name by you before saving, which allows it to be recognized again.

All files have an three-letter »extension« after the name which identifies them to you and the computer as being of a particular type. File extensions are always shown with a »«, before them, and we use the convention of placing a »*« before the »«, to mean »any filename with that extension«. The most common are:

*.SON a global file containing all SONG data
*.BON a »backup song«
.PAT a whole pattern (.BAT: »backup pattern«)
.SEQ a single track (.BEQ: »backup track«) (»SEQ« = »sequence«)
.MID a Standard MIDI File (.BID: »backup MIDI File«).

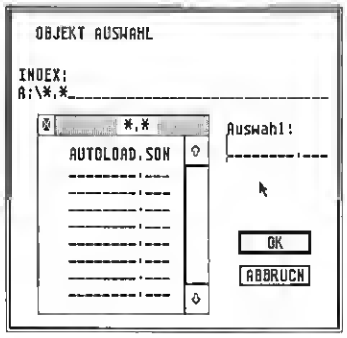
Other file extensions include *.REF/*.BEF for the »Sync Reference« *(see Chapter 27 »Hardware Peripherals: ...«, section 11)* and *.PRT/*.BRT for a printer adaptation *(see Chapter 14 »Score Printout«, section 7)*.

Never remove a disk from the drive while the »drive light« is illuminated – to do so would be to risk losing information on the disk, or risk damaging the drive mechanism.

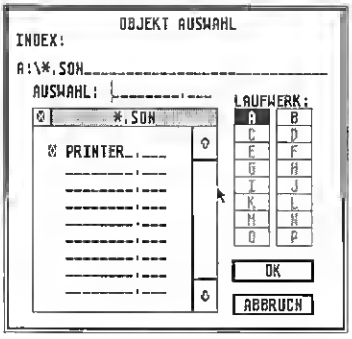
1.1 The Atari File/Item Selector

During all disk operations a dialog box appears, called an »Item Selector« or »File Selector« depending on your computer version.

The Atari File/Item Selector, which is pretty basic, can be replaced by »Accessory« programs which allow far more operations to be conducted from the Selector, though the later Atari TOS 1.4 Selector at least allows the disk drive to be selected easily.



TOS 1.4:



The Selector is part of the Atari's operating system and is therefore identical whatever type of file you are saving/loading. In the left half of the Selector is the file list, which shows files that are on the disk in the current drive. The list always defaults to showing *only files of the relevant extension*: if you click »Save SONG«, the Selector will show *.SON files only, even though the disk may contain many other types of files, including *.BON files. Folders on the disk are also shown – see below.

Additionally, the program names the operation you are doing at the top of the screen: always ensure this is correct before going ahead.

The list can be scrolled to show more names if necessary: click the »down-arrow« in the bottom right corner.

When loading, a file is selected by finding it, then double-clicking it in the list; or clicking it once so that it appears in the »Selection« line, then clicking »OK«.

When saving, type the filename in the Selection line then click OK. So long as the list is showing the correct extension in the grey bar above, the extension does not have to be typed again (though you may like to do so to play safe). All characters are in capital letters (no need to press ‹Shift›). Spaces are not allowed, but you may use ‹Shift hyphen› and ‹Shift colon›. If the name is shorter than eight characters, type the ‹.› after the last character and the cursor jumps to the extension part.

If your song, pattern etc already has a name, this name will automatically appear in the Selection line.

If the disk contains »folders« (which contain more files), these will also be shown in the list, whatever the file extension. A folder has a special »diamond« symbol before and to the left of its name. Click the folder to see what files and other folders are inside it.

The »Directory« or »Path« line above the list is very important. It always shows the »path« which you have followed to get to that position on the disk which the list is showing. The Directory line is not as confusing as it looks:

- The first letter character shows the current drive. The drive letter is always followed by a »:« which means »drive«. So, »A:« is the in-built drive A.
- The next character is the backslash »\« which means »what follows is in the »root directory« (ie what is immediately visible and not in a folder)«. If there is another »\« after a name, *whatever is between the backslashes is in a folder.*
- After the »\« comes either the folder name if open, or the »*«, which means »I am displaying files of any name«.
- Lastly, the extension is shown as a »«. followed by three characters. Instead of the three characters, there may be a »*«, meaning »I am displaying files of any extension«.

The Directory line can he directly edited: click it so the Selector cursor moves up, then use ‹Esc› to delete the whole line, ‹Backspace› or ‹Delete› and the cursor arrows. With the above knowledge, you can request your own »path«. If you type:

```
A:\*.*
```

you are saying: »show what is on the drive A disk, whatever the filename, whatever the extension (ie show everything). If you type:

```
C:\PRINTER\*.PRT
```

you are saying: »show what is in the hard disk partition C, in the PRINTER folder, whatever the filename, but only files with the extension ».PRT««.

To *carry out* the command, click in the grey bar above the list. To close folders, click the list's »close box« in its top lefthand corner.

Directory requests may be carried out whatever »File« menu function you clicked: if you clicked »Load SONG«, but wish to see all the files on the disk, use ‹Backspace› to delete the .SON extension and replace it with .*, followed by clicking the grey bar (this is how to find and load a *.BON file, for instance).

More recent Atari ST Selectors have additional drive boxes which you click to select a drive, and sophisticated Selector accessory programs have pre-programmed extension boxes, too.

For more information see your Atari ST operating manual.

2. »Save/Load SONG«

To save a song to disk, click »Save SONG« (»File« menu) or press ›Alternate-S‹. To load a song from disk, click »Load SONG« or press ›Alternate-L‹.

These two functions are the two most important »File« ones, as they save/load the entire contents of the program.

Always save your work using »Save SONG«, even if you have only used one pattern: as well as all the music, this saves the tempo, time signature, MIDI Thru options, »Transform« sets, »Sync Reference«, Hyper Edit sets etc etc.

Save your work regularly to disk, especially after recording or editing music that you would hate to lose if there was a power failure.

A loaded song will replace (delete) the current one in the RAM (or its current version), so ensure this is what you wanted to do before clicking »OK«.

If you save a song, and the song's name is already displayed in the »Selection« line (because you previously loaded or saved the song), click »OK«. If a file with the same name and extension is already on the disk, the program will tell you that a file by that name exists, and will ask you whether you want to rename that file on disk so that it has a .BON extension or whether you want to delete the file on disk.

Always respond by clicking »RENAME«: this keeps the file already on disk (but gives it a .BON extension), and saves the current one with the .SON extension. That way, there is always a safety backup on the disk, allowing you to go back one version if you made a mistake.

(You could give a new name to the same song each time you save it, thereby keeping each version (eg »MYWAY1.SON«, »MYWAY2.SON«, MYWAY3. SON etc) but you would soon run out of space on the disk.)

If the computer alerts you to the fact that there is no space left on the disk, you should either save onto a fresh formatted disk, or delete a file *(see section 7 »Delete File«)*.

2.1 Backup disk copies

There is no guarantee that your disks will not corrupt and lose data while in storage.

Golden rule: if you work with computers, always make copies of your important disks and store them separately.

A few seconds spent making a backup copies of your files on separate disks at the end of the working day is time well spent.

There are many reasons why a file can sometimes not be read by a program, but following the advice in this manual should minimize the risk of avoidable file corruption.

Rarely, if a file cannot be accessed, it may be that the disk drive which saved it was out of alignment. It reads its own disks, but not those of other computers; and other computers cannot read its disks.

Do not be tempted to use »unbranded« disks because they are cheaper. This is a false economy as they tend to be less well examined by the manufacturer than well-known brands.

2.2 Handling disks

Disks are sensistive to magnetic, mechanical and thermal influences: loudspeakers, power supplies, dust, babies, Coca Cola etc. Carry them in special disk boxes or wallets and do not leave them lying around in the sunlight or on top of the computer.

2.3 »Autoload Song«

There is a method of defaulting the program with a user's favourite parameters and functions when it loads.

If a file called »AUTOLOAD.SON« is present *on the program disk* it will be auto-loaded whenever Creator or Notator are started.

Hard disk users: AUTOLOAD.SON must be in the same partition/folder as the program.

To create an AUTOLOAD.SON: load the program, set your defaults *(see throughout manual for details)* such as Channel names, Fonts, MIDI Click etc, ensure the program disk is in the drive, select »Save SONG«, type AUTOLOAD.SON in the »Selection« line and click »OK«.

Notator and Creator may come with an AUTOLOAD.SON already on the disk: this can be edited to your configuration and saved again. Do not use any other name than »AUTOLOAD.SON«.

2.4 Multitasking disk operations

You may save and load a song to/from disk while another one is still playing; useful in live situations.

When loading while another song is playing, a dialog box appears saying »PRESS OK FOR SONG EXCHANGE«. When you are ready to exchange them, click »OK« or press ›Return‹: it takes a couple of seconds to swap songs. The current song is deleted, and the new one installed in the RAM. Song swapping only works if there is enough RAM to hold two songs.

To save a song while the sequencer is in »Playback« mode: same instructions as for saving a song when stopped *(see section 2 above)*.

Disk formatting is also possible during »Playback« mode.

The screen display freezes during these operations: this has no effect on the song being played, which carries on as normal, though you lose the MIDI Thru function and access to the controls.

2.5 Pre-programmed »Load SONG«

Inserting and naming a »P_USER 20« event in the event list will automatically load another song into the buffer while the current one is playing; useful in live situations.

The song is not actually loaded into the current RAM, but waits in the buffer for a song exchange.

– Drag a P_USER event into the event list from the »PSEUDO« box on the left. Place it at the desired time position.

– Alter its »first data byte« to 20, its »second data byte« to 0.

– Press ›Shift-Esc‹ to open a naming line: type in the song's filename (lower case letters, no spaces, no extension), click »OK«.

The name will not be seen unless the »#« icon is enabled. To edit a name, press ›Esc‹ when the event list cursor is on the event.

When the song reaches the P_USER 20's position, the program will look for the named song *in the disk drive (floppy or hard) and folder last accessed* and load it. If a »file not found« message appears, the program is probably looking in the wrong folder: check where the song is under »Load Song« and see that the correct folder is open. The program will not load songs if you are in the event editor.

A second P_USER 20 later in the event list, whose »second data byte« is 1, will automatically exchange the two songs and start the new one from the beginning.

Or, if the second P_USER's »second data byte« is 0 again (no need to name it), the songs will be exchanged but the program will wait for a manual start command.

Or, if there is no second P_USER 20 event, the songs can be manually exchanged with a right mouse button click (mouse pointer is not visible) or pressing any key but ›Return‹ or ›Enter‹: the current song stops and is replaced with the new one. A left click or pressing ›Return‹ or ›Enter‹ cancels the operation: the current song continues playing and the new song is deleted from the buffer.

3. »Load System«

»Load System« (»File« menu) allows the importing of various settings and parameters from *.SON files on disk into the current song without affecting its arrange list, patterns, tracks, events.

The window allows the selection of one or more options. Click »LOAD«, and the usual File Selector appears. In the list, double-click the *.SON file whose parameters you wish to import.

Channel Names see Chapter 6 »Tracks«, section 3.1.1.

RMG MIDI Definitions the user-definable faders, switches etc, see Chapter 22 »Realtime MIDI Generator«, section 11.

RMG Snapshots see Chapter 22 »RMG«, sections 7/8/9.

Transform Parameter Sets see Chapter 24 »Transform«, section 2.2.

Universal Maps see Chapter 24 »Transform«, section 6.

Drum Map see Chapter 11 »Score Display«, section 26.

Notepad see Chapter 21 »Overall Program Functions«, section 8.

Hyper Edit the parameters of eight Sets each with sixteen Instruments, see Chapter 15 »Hyper Edit«, sections 4 and 6.

Font Settings Notator only, see Chapter 13 »Text Functions«, section 6.

Adaptive Groove Parameters the eight Sets, see Chapter 17 »Quantize, ...«, section 3.

Key Macros see Chapter 3 »Concepts«, section 2.1.

4. »Save/Load Sequence«

To save a track to disk, click the track then click »Save Sequence« (»File« menu). To load a track from disk, click an empty track location then click »Load Sequence«.

For details, read section 2 »Save/Load SONG«, replacing references to the song with those of a track.

Saving a track has a number of uses, storing SysEx, ideas etc.

5. »Save/Load Pattern«

To save the current pattern to disk, click »Save Pattern« (»File« menu). To load a pattern from disk, click »Load Pattern«.

For details, read section 2 »Save/Load SONG«, replacing references to the song with those of a pattern.

A pattern can be loaded into any empty pattern window.

Saving a pattern has a number of uses, especially where you wish to import a music section into a different song. If you are not doing this, however:

Always save work as a .SON file.

6. Standard MIDI Files

6.1 Standard MIDI File formats

Standard MIDI Files are a more-or-less-standard data format which are designed to enable data created on different manufacturers' MIDI sequencers to be transferred between programs. Creator/Notator support Format 0 and 1.

Format 0 one track containing an unlimited number of events on all sixteen Channels.

Format 1 for several parallel tracks containing an unlimited number of events on all sixteen Channels.

Only information that is in one pattern whether it contains one or sixteen tracks is saved as Standard MIDI File.

Creator/Notator will *save and load* in the SMF format:

- the full high resolution of 1/768th notes
- all MIDI Channel data (Notes, Pressure, Control ...)

- SysEx messages without Handshakes
- Track name (handled as first text event in the track)
- Text events
- Tempo
- Time signature

The following information cannot be recognized by a Standard MIDI File:

- any track parameters (Transpose, Velocity ...). Therefore, use »Normalize« and »Process Data« to alter the events *(see Chapter 6 »Tracks«, section 4).*
- Channel and pattern names.
- any P_USER events (except for Tempo and Text)
- Notator's additional graphical symbols (slurs, accents, dynamic symbols ...) and page layout. Beware that the user-rests are notes, so delete them before saving as a Standard MIDI File.

Note: the data format of Standard MIDI Files is not fully developed, so there may be incompatibilities.

6.2 »Save/Load MIDI Files«

To save the current pattern to disk in the Standard MIDI File format, click »Save MIDI File« (»File« menu). To load a Standard MIDI File from disk, click »Load MIDI File«.

The current pattern is saved as Format 1. If it contains only one track, this track is saved as Format 0 (some sequencers can read only Format 0).

Standard MIDI Files should be used *only* when transferring files to or from other manufacturers' sequencers.

For details, read section 2 »Save/Load SONG«, replacing references to the song with those of a MIDI File.

7. »Delete File«

To delete a file on disk, click »Delete File« (»File« menu).

This deletes only what is already on disk, not what is in the computer's RAM. The File Selector displays every file and backup file. A message will ask you whether you wish to proceed with the deletion: if »OK«, the file is gone forever!

8. »Format Disk«

To format a disk, click »Format Disk« (»File« menu).

Before first use, a new disk must be »formatted« by the computer to conform to the Atari way of storing information. This erases any data that might already be on the disk, so double-check that you want to format it.

The dialog box asks whether you wish to format »SINGLE« or »DOUBLE«: always click »DOUBLE« as this maximizes use of the disk (both sides).

Do not attempt to format disks that were once formatted and used on other devices (samplers etc). Always keep a good supply of branded, formatted disks available.

Chapter 29

MIDI Multitasking with Soft Link

29

1. The basics

In furtherance of a »Universal System«, the aim of Soft Link is to open the architecture of Notator and Creator to other programs, so as to encourage the creation of many different applications integrated with hardware peripherals.

To fulfill the differing requirements of users and produce results as fast as possible, there are three Soft Link Levels:

Level 1 the Director desk accessory represents the simplest form of Soft Link. It works without having Notator or Creator in the same computer, with programs in a one Megabyte ST computer.

Levels 2 and 3 the Soft Link manager is an integral component of Creator SL and Notator SL and allows up to nine programs to simultaneously reside in the computer's RAM, so long as there are at least two Megabytes of RAM (the ideal being the maximum four Megabytes). If you use the Atari SLM804 laser printer then four Megabytes become necessary when running Soft Link as well.

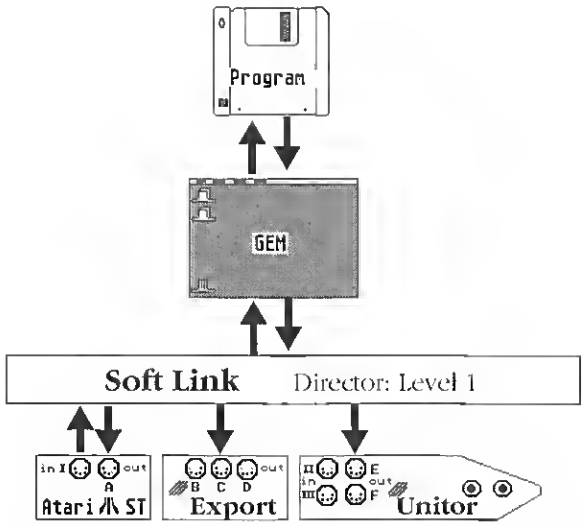
2. Director

2.1 The basics

Director, a desk accessory which runs without the need for Creator or Notator to be present, is copied onto other programs' disks and redirects the MIDI data of these programs to MIDI Outputs B to F in Export *and* Unitor.

Director is available as a separate disk only: it is not provided with Creator SL or Notator SL because it is not for use with these programs. If you want to use Director with your other programs, please request the disk from your music dealer.

It allows the user to make use of the additional ports of his/her hardware when using other programs, avoiding the need to re-patch instruments to the Atari MIDI Output.



Director represents the simplest form of Soft Link; it uses very little RAM and so can be used with one Megabyte Atari's as a MIDI patcher. As such it provides a link between other programs and the MIDI Outputs of the hardware. Since Director simply expands the existing MIDI routines in the Atari operating system, it maximizes the potential of compatibility (providing, of course, that the programs also use the Atari operating system's standard MIDI routine).

See section 5 on »Compatibility«.

2.2 Director as »Accessory«

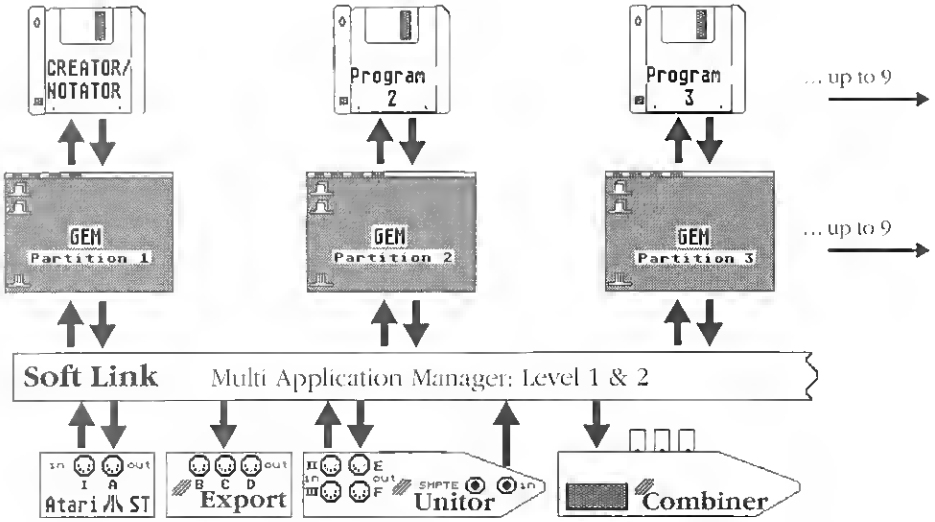
The Director is available separately, not with Creator/Notator SL, and comes with instructions on loading it onto other programs' disks. Please refer to those instructions.

3. Soft Link: the basics

Soft Link switches between up to nine simultaneously-residing programs in the computer's RAM.

The programs behave as if they are each running on a separate computer, each with its own (software) MIDI Ins and Outs.

Notator/Creator SL act as a central control for MIDI Ins and Outs, merge and transmit all the data, and synchronize to SMPTE. These functions remain active, even when other programs are running (»Multitasking«). MIDI Thru processing and realtime processing (eg realtime »ghost tracks«, filtering or »Transform«) of all three Inputs make the system universal in its potential.



When Soft Link is installed, the computer's memory will be divided up into user-definable RAM partitions. Each partition then becomes a virtual computer, ready to load its own set of instructions.

Not every program is Soft Link compatible. To be safe, *before* loading a new program, save any important data you may have in other partitions.

Please see sections 3.2 »Switching partitions« and 5 »Soft Link and compatibility«.

If using Combiner, Soft Link auto-switches between the four dongle slots as you switch between the first four partitions. Unitor remains available at all times.

This way, key-protected programs remain fully integrated; if, say, you select X-Alyzer in partition 2, its dongle in Combiner port 2 is auto-activated; Unitor's MIDI Inputs and Outputs and SMPTE synchronization remain fully operational.

3.1 Starting Creator/Notator from Soft Link

Double-click »SOFTLINK.PRG« to start Creator SL or Notator SL.

After a while you can start work. If instead a »not enough memory« message appears, *see section 4 »LINKEDIT« below.*

Soft Link is modular: if you do not want to use the Soft Link part, or cannot due to lack of RAM, you can start Creator or Notator without Soft Link, and no extra memory will be used.

To start Creator/Notator without Soft Link, double-click their program icons.

3.2 Switching partitions

Press the *right* ›Shift‹ key and one of the numbers in the calculator keypad to select the partitions. ›Right Shift-1‹ selects Notator or Creator, higher numbers select other partitions where available.

The partition number can always be seen at the top left of the screen.

The currently-displayed program continues its functions from the point it was left at when it was last accessed. Creator or Notator, however, will continue running even if another partition is displayed, together with all its realtime functions such as Record, CYCLE, MIDI Thru, realtime »Transform«, realtime »ghost tracks«, realtime »Playback« parameters, »SMPTE Sync« mode etc. If these tasks are a heavy load for Creator, then the reaction times of the other program will be a little slower.

The user-interface was designed to give the impression that more than one computer was being used, such that it seems that several computers are being run with the same monitor: switching is instantaneous between partitions. Each partition has its own, virgin desktop from which to start the desired program; Soft Link allows you to operate the Atari as normal.

The following commands remain active in any partition:

Calculator keypad:

›Right Shift-0‹ – Start
›Right Shift-.‹ – Continue
›Right Shift-Enter‹ – Stop
›Right Shift-*‹ – Renew Record (for safety, you may not *start* recording when not in Creator/Notator SL).

Do not switch partitions while saving to/loading from disk.

Although it can be safe to do so, it is wiser to ensure the safety of your data by remaining in the partition. The Atari does not have a »Disk/File locking system« that you find in a large office multi-user system, and data can go adrift if you switch programs while valuable data is being saved.

Not every program is Soft Link compatible. To be safe, *before* loading a new program in a partition, save any important data you may have in other partitions.

In certain cases, a non-compatible program could cause the system to crash. Pressing the computer's rear panel »reset/warm start« button *once* activates an »emergency« function which will attempt to rescue the data in the RAM and will switch to another partition: you now have a chance of saving the other partitions' data to disk. Do not switch back to the »bad« partition: once you have saved any data, power down the computer, wait twenty seconds and restart the Soft Link system.

3.3 Manipulating MIDI Data transmitted from the partitions

MIDI data from the partitions is internally mixed with the Creator/Notator MIDI data. The MIDI Thru function's

»Playback« parameters (ie the track parameters set in an empty track) will operate as if the data were coming from outside the system.

Track parameter »CHANNEL« is particularly important since this is where you select the output ports A to F for the various partitions' MIDI data.

In Record mode, Creator/Notator will record and display other partitions' MIDI data. Creator/Notator's »Soft Link« filter window (»Options« menu) allows you to filter out unwanted data before recording.

This special record filter will affect the recording only. All MIDI data is transmitted anyway, unless the MIDI Thru function's Output Filter (»Output Filter« in the »MIDI« menu) is realtime-filtering data.

The MIDI data present at all three MIDI Inputs (Atari and Unitor) is available to all partitions in a merged state.

SysEx data arriving at the Inputs will be re-transmitted by the non-Creator/Notator partitions to the Outputs only if »SysEx THRU« is enabled in »Soft Link« (»Options« menu).

The disabling of this »SysEx THRU« option represents an additional SysEx output filter for the use of non-Creator/Notator partitions only, saving you the need to manually switch off the MIDI Thru function. For example, if incoming SysEx data intended for an editor/librarian program was re-transmitted, this would cause problems.

Example: one obvious application for Soft Link is combining Creator or Notator with one or more MIDI editor/librarians. Switch to a new partition and start the desired program (from now on, accessing this new program is done via a keystroke as discussed above – no need to keep loading the program).

The SysEx sound data of this editor can be recorded by Creator/Notator as if it had been »ported across« from a second computer: in Creator/Notator, click »RECORD«, switch to the desired partition, select and change sounds in the editor, cease recording via »Right Shift-Enter«, switch back to Creator, and you will find the editor's data in the Creator track, which can be played back. Editors often send additional non-SysEx MIDI messages (Controllers or Pitch Bend) which you can filter out in the Soft Link Record Filter (»Options« menu).

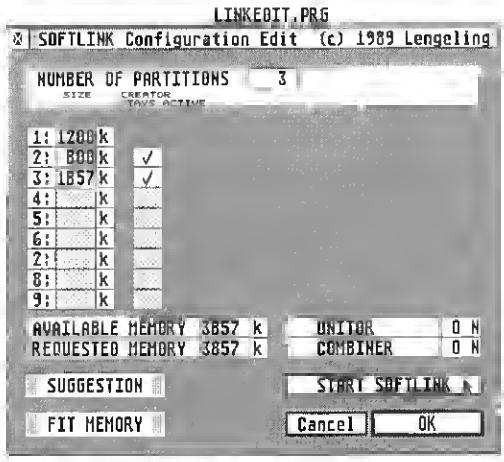
4. LINKEDIT: editing the Soft Link configuration

Before starting Soft Link for the first time, double-clicking »LINKEDIT.PRG« on the program disk allows you to define RAM allocation, number of partitions and hardware (Unitor or Combiner).

If Notator SL or Creator SL are already running, you have to power down before accessing »LINKEDIT.PRG«.

Do not omit to define your hardware configuration. Clicking »OK« or »START Soft Link« will store all the details.

The last partition defined (the one with the highest number) will receive any spare memory not yet allocated, so the size value should be seen as a minimum. If there is not enough memory left for that partition, LINKEDIT will automatically end the process after a warning notice and store the details.



Explanation:

Creator Stays Active determines whether Creator remains active when that partition is active. You should switch this off for a partition containing a program that is not Soft Link compatible to avoid causing problems to the system.

Available Memory shows the number of K-bytes free in the RAM.

Requested Memory shows the K-bytes used by the partitions.

Fit Memory fits the request for RAM according to what RAM is available by adapting or removing the last partition.

Suggestion suggests a way of distributing the memory to all requested partitions, which you can of course alter.

The last two functions do not make sense if used after Soft Link has been started because LINKEDIT can only recognize the overall memory that is available at a new start, not the available memory in current partitions. How many kilobytes per partition you need depends on how much you think you will need for the particular program you are using. The minimum size is around 300 KB. In partition 1, Creator should have at least 900 KB and Notator at least 1200 KB (unless you are using the Atari SLM 804 laser printer, in which case at least 2200 KB should be reserved).

As a rule, fewer but larger partitions are recommended: the remaining ones should receive around 1100 KB. Graphically-oriented editor programs usually need quite a lot of RAM, pure text programs such as Word Plus need less. Accessories should be used with care since they use up valuable memory.

4.1 Examples of configurations

	Partitions total number	1	2	3	4
MEGA ST 2/Creator	2	1100	800		
MEGA ST 4/Creator	3	1400	1200	1200	
	4	1100	1000	1000	700
MEGA ST 2/Notator	2	1200	700		
MEGA ST 4/Notator	3	1500	1200	1100	
	4	1200	1000	1000	600
MEGA ST 4/Notator/ Atari Laser printer	2	2500	1300		
	3	2200	1000	600	

5. Soft Link and compatibility

Compatibility is vital to a pan-program system such as Soft Link; since common hardware (Atari) is used, it is important that a common standard be adopted. There are two ways of approaching this question: you either make use of an established standard, or attempt to introduce a new one. With an established standard, compatibility is by definition, obvious. A completely new data format, on the other hand, once established, has the advantage of offering a fresh solutions to new requirements.

Soft Link, with its three-Level hierarchy, combines the best of both worlds through evolution rather than revolution. Levels 1 and 2 use, modify or replace the existing Atari operating system to be found in every ST computer: that way, programs that already conform to this existing standard are immediately compatible, without any need of modification. Level 2 is already capable of fulfilling most demands required of a pan-program system, such as Multitasking and the internal routing of MIDI. In principle, therefore, all kinds of editor programs are capable of running in Soft Link, without any particular limitations.

Trials have shown that at the time of going to print, almost every program tested started and could be operated in Soft Link. Non-MIDI programs such as word processors and graphic drawing programs present no problems. The MIDI Output of many MIDI programs works without a hitch. What became clear, though, was that there were frequent problems of compatibility with MIDI In routines. These programs need a (normally very small) modification to get them to make use of the existing operating system *(see the section for Programmers below).*

This general problem was touched on above: one computer is used by several programs, and it is clear that the hardware components themselves cannot be physically duplicated. Conflicts can therefore appear if programs reserve individual parts of the hardware for themselves, or if there is no operating system routine that can coordinate a quasi-simultaneous usage. A graphic example would be two watch faces which both run off the same mechanism, but which both have to tell a different time and run at different speeds. The problem is basically solvable so long as the requirements regarding the display of these two times of day are met by a negotiated solution.

This watch face example is similar to the problem which confronts Soft Link: no matter how precise *within itself* a type of software that generates its time axis from the computer hardware is, it does not imply that it will be immediately cooperative with other programs of the same type. We are here mainly talking about software with sequential Hfunctions, such as MIDI sequencers, mixdown automation, algorithmic composers etc, or non-MIDI applications such as modem programs.

The internal synchronization of these sorts of programs can be made to happen through the use of special internal formats and routines only. In order to make any compatibility applicable as universally as possible (ie not just for the products of one manufacturer), a *common approach* must be adopted as regards the development of a future standard. Soft Link Level 3 already offers routines that enable, amongst other things, internal synchronization. The aim is to unify the data protocol of as many manufacturers as possible, to adjust and adapt the internal data format and thereby to bring about a universal standard. At the time of writing, all this is just starting, and is bound to take time to come to fruition.

6. Programmers: Welcome Aboard!

Wherever possible, Soft Link uses, alters or replaces the Atari system calls. When calling these routines (some BIOS/XBIOS traps) you are in fact calling Soft Link which expects the same parameters as the original routines. This means that programs that are capable of running with the original operating system can run with Soft Link without any special modification.

Soft Link Level 1 accessory Director alters the following system routines: Bconout (3); and Midiws0);. Depending on the routing selected in Director, MIDI data is sent to the physical ports original Atari (A), Export (B ,C, D) or Unitor (E, F).

Soft Link Level 2 is built-into Creator SL and Notator SL. Several partitions with standard desktops including GEM, VDI, GEMDOS are completely emulated. Programs can be started in each partition. Partitions are changed by Soft Link without the need

for quitting or reloading programs. All realtime functions of Creator SL are active all the time (Play, Record, MIDI Thru, SMPTE Sync, MIDI communication).

These operating system calls are replaced: Bconout(MIDI); Midiws(); Bconin(MIDI); Bconstat(MIDI); and Xbtimer(TIMER_A).

The MIDI data of both transmit routines is internally and logically correctly merged both with the sequencer's transmit data and any data that arrives at the MIDI Inputs; it is sent to the MIDI Outputs according to the selected routing and internally displayed where appropriate. Both MIDI receive routines take account of the already logically-correctly mixed data stream of all physical Inputs (which includes Unitor) and pass them to the caller. The MIDI interrupt vectors may not be altered.

Creator SL and Notator SL need Timer A. Timer A is therefore NOT available to other programs. If a program installs a Timer A interrupt routine via a Xbtimer(), Soft Link will periodically call this routine using the same stack structure as it would if called by Timer A directly.

Timer B is not used by Soft Link. Timer D is used for RS 232-Export (baud rate). You may install routines in VBL slots.

Soft Link Level 3 offers new functions which are independent of the Atari operating system. Write to EMAGIC for more information.

There is a simple test to see if Soft Link (Level 2 or 3) is installed in the computer (in »C«):

```
if (xbios(27778) == 15200259L) /* Yes, Soft Link is there! */
```

It is not absolutely necessary to check for Soft Link's presence. If it is there, you may draw the following conclusions:

1. It is not necessary to increase the TOS MIDI buffer. No harm comes of doing so, but it wastes memory.
2. MIDI Thru functions should be completely switched off, to prevent a multiple output of data.

Appendix A

1. Some Creator and Notator concepts

1.1 »Playback« versus »event-altering« functions

Creator and Notator functions can basically be divided into two types:

1. »Event-altering« functions that alter events in the computer's memory. Because of this alteration, the result of using an »event-altering« function can be seen in the event editor. The result of using an »event-altering« function can be reversed by using the function to do the opposite, but sometimes even this is not possible; it depends on the function, on its complexity, and on the scale of the alteration that was made.

   Most »Event-altering« functions are available as menu options *(see Chapter 6 »Tracks«, section 4 »When track parameters can be »event-altering««, Chapter 18 »Copy, Merge, ...«, Chapter 19 »Event-Altering Functions«, Chapter 24 »Transform«)*. And naturally, when you use the event list and any of the editors, you are »altering« events.

2. »Playback« functions that do not alter events in the computer's memory, but alter them as they pass, »live«, through the MIDI Outputs when they are played. Because of this, the result of using a »playback« function cannot be seen in the event editor (the event editor can only show what is in the memory).

   »Playback« functions have the advantage of being immediately reversible, with the guarantee that the original data has not been touched.

   As many functions as the computer allows are »playback« functions. Every visible value on the main page is a »playback« function *(QUANTIZE and GROOVE excepted – see Chapter 17 »Quantize, ...«)* including the arrange mode. The overall ease of handling of data in the arrange mode and the resulting reduction in the use of memory are other advantages of »playback« functions *(see also Chapter 6 »Tracks«, Chapter 20 »Arrange Mode« and Chapter 23 »Realtime MIDI Functions«)*.

1.2 What can be recorded?

All MIDI Voice (Channel) messages (Note On, Note Off, Pitch Bender Change, Channel Pressure, Poly Pressure, Control Change, Program Change) and System Exclusive *(see Chapter 8 »Types Of Event«).*

MIDI System Realtime and Common data cannot be recorded but is received, processed and transmitted (MIDI Clock, START, STOP, CONTINUE, and SONG POSITION POINTERs etc).

1.3 Automatic Timing Control and »Overflow«

The greatest advantage of MIDI over previous interfacing systems lies in its ability to interface diverse musical devices with relatively few problems, and in its wide acceptance among musicians. MIDI is a very fast sequential computer interface, although for certain applications not fast enough.

If a MIDI set-up includes a lot of instruments where a vast number of notes and other events are playing (especially where pitch, modulation and pressure information is involved), the limitations of the MIDI protocol start to be noticed.

The best policy is therefore to manage data as efficiently and carefully as possible. Even under the heaviest of loads, however, Creator/Notator will not be beaten. The correct data sequence is *constantly* analyzed: the program independently sets priorities and plays all data with the greatest possible speed. If the program has to transmit vast quantities of data, then there may be tiny delays in computer keystroke reaction times or in the screen display.

The »Overflow« box (below the pattern window) is a visual way of monitoring any discrepancy between the strict internal clock and the *actual* time position of the note (the »should be« and »actually is« times). The actual time position will always catch up with the »should be« time position as soon as any discrepancy has appeared. These discrepancies can occur, for instance, when a SysEx message of some length is inserted *within* a playing piece (which should be avoided if possible). Short blips in the box can be ignored. If the beam goes speckled, there was probably a perceptible delay, but this only happens if you are overburdening the system.

1.4 »Play Algorithm«

The »Play Algorithm« function optimizes the timing in the playback of existing events by ensuring notes get priority if ever there is the possibility of any delay during a heavy load. It should always be left switched on, in the »Flags« menu.

1.5 »Data Reduction«

Additionally to the Automatic Timing Control and Play Algorithm functions, the program conducts a reduction of data after independent analysis of certain musical criteria. By carefully removing certain non-note data, it ensures that the result will not be heard, but that the load is taken off the MIDI system.

For certain applications, »Data Reduction« («Flags« menu) may need to be disabled, eg certain mixing desks use MIDI to control muting, and all the data must be present.

1.6 Clock »Interpolation« in »MIDI Sync« mode

The standard MIDI Clock's resolution is just 1/96th note, but the program is able to retain its much higher 1/768th (1/1536th) note resolution in »MIDI Sync« mode.

This is done by introducing or »interpolating« its own Synchronization reference points between the incoming Clock pulses: the program constantly measures the distance between two received MIDI Clock pulses, calculates the tempo, and by setting its own 1/768th pulses between the MIDI Clock's pulses, can transmit notes very accurately at the required time positions.

1.7 One note – two events

The MIDI protocol defines a note consisting of two separate events: a Note On event and Note Off event. Although this division is very handy for a computer, it does rather contradict the musical idea of a »note«.

However, Creator/Notator as far as possible give the impression of the note being complete. Every function including the copy functions include the Notes Off without input needed from the user: Note On events always include the Note Off event. This may clearly be seen in the event list, where the program automatically calculates the difference in the time positions between the Note On and Note Off and displays the difference (the note's length) in bars, beats etc. Whenever you change a note's length, what you are doing is in fact moving the Note Off event in time. So you never need to see Note Off events, which is why they are filtered out of the event list display.

The division between the two events does not usually cause any confusion. The two events are related by their common MIDI Channel and common pitch. This relationship is perfectly clear so long as a Note On has a corresponding Note Off at some point; this will always be the case when you are playing a keyboard because (think about it!) you cannot play a key a second time without having let go of it first.

This situation changes when you hook up a MIDI sequencer, because a sequencer will allow you to do anything by editing and the use of complex functions. The flexibility

of being able to manipulate, for example, single Note Off events means that the situation could arise where you have moved a Note Off to a point where you have two Note On events sharing the same MIDI Channel and pitch, followed by two Note Off events.

This situation is not covered by the MIDI protocol: do the first Note On and first Note Off belong together, and the second Note On and second Note Off belong together, or is it vice versa? Or does the second Note On nullify the first Note On, or will the second Note On just be ignored? Different MIDI devices react differently in this grey area. In practice the aural effect of such a situation is difficult to spot, since the musical changes that ensue are slight and normally amount to a »legato« effect where none existed before.

The program's event editor recognizes the above situation and prevents it from arising in the first place. You will not be able to increase the duration of a note beyond a certain length: this maximum length is determined by the point at which increasing the length any further would cause the Note Off to overlap an existing Note of the same MIDI Channel and pitch.

1.8 Resolution: display in notes or pulses per quarter note?

In Creator/Notator we talk about the smallest unit of time being a 1/768th note or 1/1536th note. Some people prefer to talk about resolution in terms of »pulses per quarter note« or »ppq«: they divide the whole note by the four quarter notes.

So, in these terms, Creator/Notator's highest resolutions are 192 ppq (1/768ths) switchable to 384 ppq (1/1536ths). MIDI Clock's resolution is 24 ppq (1/96ths).

1.9 Values: »Inclusive/exclusive«

Everywhere in the program, without exception, the beginning (or »lower«, »left«, »from«) value of a zone or segment always includes that value in the calculations: the value is said to be »inclusive«.

The end (or »upper«, right«, »to«) value is never included in the calculation: the value is said to be »exclusive«.

Example 1: »copy a segment from »1 1 1 1« to »5 1 1 1«« means »copy the first four bars« (the »to« value is not included).

Example 2: the Transform condition »delete all notes from C3 to C4« means »delete all those notes but not the C4's«.

2. »Kill Score«

shift-Q

Pressing »Shift-Q« on the main page opens a dialog box, asking whether you wish to »Kill Score Program«. Clicking »KILL« gets rid of the score editor without harming any data. Use this in an emergency if you need more memory space. To retrieve the score editor, reload the complete program.

3. Notation display »Pseudo« events

58 – Guitar Tabulature.
59 – Chord Symbol
60 – Text
61 – Lyrics
62 – Dynamic signs
63 – Segno
64 – Slurs
65 – Crescendi
66 – Trills
67 – User Accents
68 – Repeat Bar
69 – Rests of more than one bar
70 – Space left of note
71 – Space right of note
72 – Space either side of note
73 – Arpeggio
74 – Tempo sign
80/81 – Clef change upper/lower stave
82/83 – Key change upper/lower stave
84 – Format Part (Page Preview)
85 – Format Score (Page Preview)

4. Notation display parameters and their effects

Effect	*Parameter*
whole song	Sloping beams »Minimum distance« Alternative 4/4 and 2/2 symbols Slur thickness
Effect	Parameter
global	Time signature
per track	Key signature Clef Display format or quantization Overlap Correction mode Rest Correction mode Interpretation mode Vocal Mode (no beams) Stem direction Stem hide Split stave/Split point Empty stave Miniature stave Polyphonic Mode Mapped Drum Mode
regional	Key signature change Clef change Time signature change
local	Enharmonic shift Note heads Beaming Syncopation display Stem direction Stem hide Graphical microshifting of notes Tuplets Force accidental Accidental distance

5. Glossary of terms used in manual

Absolute: the time shown is the correct value, it is not relative to any offset time, so no need to make mental compensations, eg »Global Positions« (»Edit« menu) shows events' time positions in the event editor in absolute terms: the times are shown relative to the »1 1 1 1« of the whole program. (See Chapters 5 »Positioning«, section 1.4 »Position/length displays in absolute units of time« and 20 »Arrange Mode«, section 13 »Global position«.)

Alert box a message which automatically appears on the screen to warn you of something, eg if you try to delete the last remaining entry in the arrange list, you are told you cannot do this.

ASCII the representation of numbers in the form of an internationally-agreed set of symbols (see Chapters 9 »Editing in the Event Editor«, section 9 and 14 »Score Printout«, section 7.5).

Byte part of a MIDI event (see Chapters 4 »Definitions«, section 1, 8 »Types Of Event« and 9 »Editing in the Event Editor«, section 4).

Calculator Keypad the block of keys on the right of the computer which looks like a calculator.

Checksum certain MIDI devices require SysEx Checksums to ensure that transmitted data is correct. The RMG does not transmit Checksums (see Chapters 8 »Types Of Event«, section 6.7 and 22 »Realtime MIDI Generator«, section 11.5).

Current as in »current track/pattern/event« etc. The item which is under the cursor.

Cursor the black rectangular box which moves about the screen highlighting values, parameters etc. For example, the track cursor shows which is the »current« track by moving up and down the pattern window; the event list cursor highlights various parts of an event, etc. Values under the cursor may be altered by using the »+/-« keys. In the score editor, the cursor makes the current object blink on and off.

Data byte, first/second a MIDI event consists of three »bytes« or parts. The »status byte« is the first of the three and contains information on the event's own Channel and its »Status« (what it is, Note, Control etc). The »first« and »second data bytes« (which, confusingly, are the second and third bytes of an event!) tell us more about the status (eg, if it is a Note, what the pitch is and its velocity). If you take time to understand this simple concept, you will find the event list and »Transform« function easy to work with. (See Chapters 8 »Types Of Event«, 9 »Editing in the Event Editor«, 24 »Transform«.)

Default the setting which the program comes set to; the »resting« value, eg the program defaults to the »768« internal clock resolution, though you may choose »1536«.

Dialog box these are produced by some functions in the screen, and they ask you to do something, eg clicking the »EDIT« icon on the main page when the cursor is on an empty track produces a dialog box which asks whether you wish to create a new track.

Event-altering a function which changes the bytes of an event in the computer's memory, eg »Process Data«. see Chapter 19 »Event-Altering Functions«, and »Appendix«, section 1.1 »›Playback‹ versus ›event-altering‹ functions«.)

External Sync synchronizing to a external source of timing information such as a »SMPTE-to-MIDI« tape synchronizer. (See Chapters 26 »Synchronization« and 27 »Hardware Peripherals: ...« .)

Format the »FORMAT« value in the »Information Bar« sets the screen quantization. To »Format« a disk is to prepare it to take information. (See Chapter 11 »Score Display«, section 9 »Quantizing the display«.)

Function a process that carries out a number of commands on a number of events simultaneously, instead of you having to change each individual event manually, eg »Transform«.

Hexadecimal calculations based on base 16, used by professional MIDI and printer programmers (see Chapters 9 »Editing in the Event Editor«, section 9 and 14 »Score Printout«, section 7.5).

Icon an on-screen representation of a switch, button etc that does something in the program.

Inclusive/exclusive (See »Appendix«, section 1.9 »Values: inclusive/exclusive«.)

Information bar the bar full of information that sits over the top of the main page and the event editor. Contains the Main Bar Counter, time signature, »Sync« mode etc.

Input also called the »MIDI In port«, this is where the MIDI cable is attached which brings MIDI data into the program.

Locator a position which marks the start or end point of a process, Cycle etc. (See Chapter 5 »Positioning«, section 5 »Cycle mode, Locators and Autolocator«.)

MIDI Thru function passes incoming MIDI data immediately back to the Output, and can simultaneously transpose, re-Channelize etc the data passing through. (See Chapters 6 »Tracks«, 23 »Realtime MIDI Functions«.)

Memory see RAM

Menu along the top of the screen is the »menu bar«, which contains »menu headings«, each of which contain a list of »menu options«, each of which carry out a function, open a function window, etc.

Mode a state, as in »when you are in »Insert mode«, any change you make to an event's time position in the event list will affect all the subsequent events' positions as well«.

Multitasking the ability of the program to do more than one thing at once, eg you may load a song while the program is in Playback mode.

Offset a time position which is relative to the beginning. *Example:* in the arrange list, if a pattern starts at bar 5 it is said to be offset by four bars from the beginning of the song.

Output also called the »MIDI Out port«, this is where the MIDI cable is attached which transmits MIDI data from the program.

Parameter a value which is part of a function or part of the program, eg a track has a number of parameters which it can use to affect the events inside it. These track parameters include »CHANNEL«, »TRANSPOSE« etc (see Chapter 6 »Tracks«). These track parameters are »Playback« parameters (see below).

Partition in Soft Link, the memory of the computer is divided up into partitions, each of which represents a separate computer, and into which programs are loaded. (See Chapter 29 »MIDI Multitasking with Soft Link«.)

Playback Parameter a function which affects events as they are transmitted from the Output without changing the bytes of the events themselves in the memory. (See Chapter 6 »Tracks«, and »Appendix«, section 1.1 »Playback« versus »event-altering« functions.)

Port the physical sockets in the computer, Unitor, Export etc to which are connected the MIDI cables, printer etc.

Pseudo events are events unique to the program which deal with certain internal processes, eg tempo, muting, score symbols etc. They are not transmitted from the program with MIDI events. (See Chapter 8 »Types Of Event«, section 7 »Pseudo (P_USER) events«.)

Pulse the internal clock's smallest unit of time. (See Chapter 5 »Positioning«, section 2 »Resolution«, and »Appendix«, section 1.8 »Resolution«.)

RAM the computer's Random Access Memory which temporarily stores all the data while you are working. The RAM's contents should be periodically saved to disk as its data is erased if power is interrupted. (See Chapter 1 »Overview«, section 3.1 »Hardware«.)

Realtime (1) a process which can be conducted while the program is running is being conducted in »realtime«, eg changing of tempo using the pitch wheel can be done in realtime without stopping the program. (2) the name given to the internal clock's highest resolutions 768 and 1536.

Resolution the degree of control the program allows over events' time positions. The program has a highest resolution of 1/1536th notes. (See Chapter 5 »Positioning«, section 2 »Resolution«, and »Appendix«, section 1.8 »Resolution«.)

SMPTE is a universally-adopted timecode which locks two systems together and tells the slave system exactly where the master system is. (See Chapters 26 »Synchronization«, 27 »Hardware Peripherals: ...« .)

Scroll to change a value with the mouse, eg to scroll a value on the screen, place the mouse pointer on it and press-and-hold a mouse button. (See Chapter 3 »Concepts, section 1 »Mouse«.)

Scrollbar/box: part of the Atari GEM environment. Various parts of the program have a scroll bar, including Hyper Edit, the score editor, the event editor page and the Item/File Selectors. The scroll bar allows you to show an earlier/later part of the function by clicking inside it or dragging the white scroll box inside the bar, where the bar represents the overall function, and the box the screen segment visible.

Segment part of a track, eg »3 1 1 1« to »9 1 1 1« is a 6-bar segment.

Status every MIDI event has a status which determines what type of event it is, Note, Pitch Wheel, Aftertouch etc. (See Chapter 8 »Types Of Event«.)

Typewriter keypad the main keypad on the computer.

VITC Vertical Interval Timecode, inserted between video frames, allowing Notator/Creator to lock to video via Steady Eye even if the video is in »slow motion« or »freeze-frame« mode.

Index

Chapters are shown in bold type, followed by the relevant section eg for »Aftertouch« look in Chapter 8, section 4.

CAPITAL LETTERS indicate the item is displayed in capitals on the screen.

(m) – menu option
(IB) – Information Bar
(TP) – Track Parameter
(K) – Key
(PM) – Parameter Mode window
(EE) – Event Editor